IBM-PC

מחשבת

מחשבת

צעדים נוספים בתוכנית
בלימוד עצמי

יחידה 2

שעונים דיגיטליים ופירמידות

BASIC-2

תוכן העניינים

	עמוד
פתיחה בצבע ובצליל	5
פרק א - משתנים מספריים	6
פרק ב - לולאות סופיות	18
פרק ג - בנית שעון דיגיטלי	25
פרק ד - שטיחים לפי הזמנה	29
פרק ה - שכפול רובוטים	40
פרק ו - אנימציה - יצירת תנועה על המסך	49
פרק ז - מסגרות לפי מידה	52
פרק ח - בנית צורות	55
רשימת מושגים נלמדים	61
תשובות	62

פתיחה בצבע ובצליל

נפתח את יחידת הלימוד השניה בצבע ובצליל...

לשם כך העתק את התוכנית הקצרה הבאה למחשב והרץ אותה:

(אל תוטרד מכך שאינך מבין את התוכנית בשלב זה)

```
10 WIDTH 40
20 COLOR 0, C : CLS
25 C = C+1
30 FOR N=0 TO 7
40 COLOR N, 7-N, N
50 SOUND 250+N*10, 10
60 LOCATE 10, 10 : PRINT "***** IBM PC *****"
70 NEXT N
80 GO TO 20
```

כיצד מתקבלים פעלולי הצבע והצליל על המסך? (אם ברשותך מחשב צבעוני)
ביחידת לימוד זו "נפצה" את המושגים והפקודות המופיעים בתוכנית זאת:

- הלולאה הסופית (FOR-NEXT)
- משתנה מספרי

נראה כי מושגים אלה יפתחו בפנינו עולם ומלואו:
החל בבניית שעון דיגיטלי וכלה בבניית "פירמידות" וצורות גרפיות מעניינות. תוך כדי מסענו זה נשתלט על כלים תכנותיים נוספים וכל זאת תוך ניצול ולימוד תכונותיו המיוחדות של מחשב ה-IBM.
יחידת לימוד זו תהווה אתגר מחשבתי ותכנותי גם לאלה מכם שכבר עברו קורס BASIC ורוצים ליישם את הידע שלהם על מחשב ה-IBM.
וכמו שכבר אמרנו ביחידה הראשונה: נסה תמיד להתמודד תחילה עם הבעיות בכוחות עצמך ורק לאחר מכך פנה לתשובות המופיעות בסוף החוברת.

באיחולי המשך לימוד מהנה ואתגרי

דני קדם ויצחק קליסקי

"מחשבת" - מערכות למידה

פרק א
משתנים מספריים

התבונן בתוכנית המופיעה בעמוד הקודם.
אתה רואה כי בהרבה שורות מופיע N. מה תפקידו של N
"מסתורי" זה? בפרק זה נפענח זאת!

☐ תחילה מחק מזכרונו של המחשב את התוכנית.
(אינך זוכר איך? - פנה ליחידה 1 עמוד 29)

☐ העתק למחשב את התוכנית הבאה:

```
10 WIDTH 40 : CLS
20 LET N=1
30 LOCATE 10, 20 : PRINT N
```

הקפד על רווח! (אחרי LET)

☐ הרץ את התוכנית. על המסך הופיע המספר ___ (1, 10).

? מה עליך לעשות בשורה 20 על מנת שהמחשב ידפיס 2 במרכז המסך?
☐ - נסה!

משימתנו תהיה לשנות את התוכנית כך שהמחשב ידפיס מספרים מ-1 והלאה, במרכז המסך. (כל מספר מוחק את קודמו), כדי לבצע משימה זאת עליך לדעת מהו **משתנה** וכיצד להשתמש בו:

☐ פקוד על המחשב:

```
LET A=6
LET B=4
```

אל תשכח רווח אחרי הפקודה LET.

(אל תשכח ללחוץ ENTER לאחר כל פקודה)

? מה יקרה עכשיו, אם תפקוד על המחשב:

```
PRINT A+B
```

☐ תחילה חשוב - ואחר-כך נסה! (אל תשכח כמובן ENTER...)

? המחשב הדפיס את המספר : ____ (10, 4, 6, 0)

כיצד קרה הדבר?

ובכן, כאשר פקדת על המחשב:

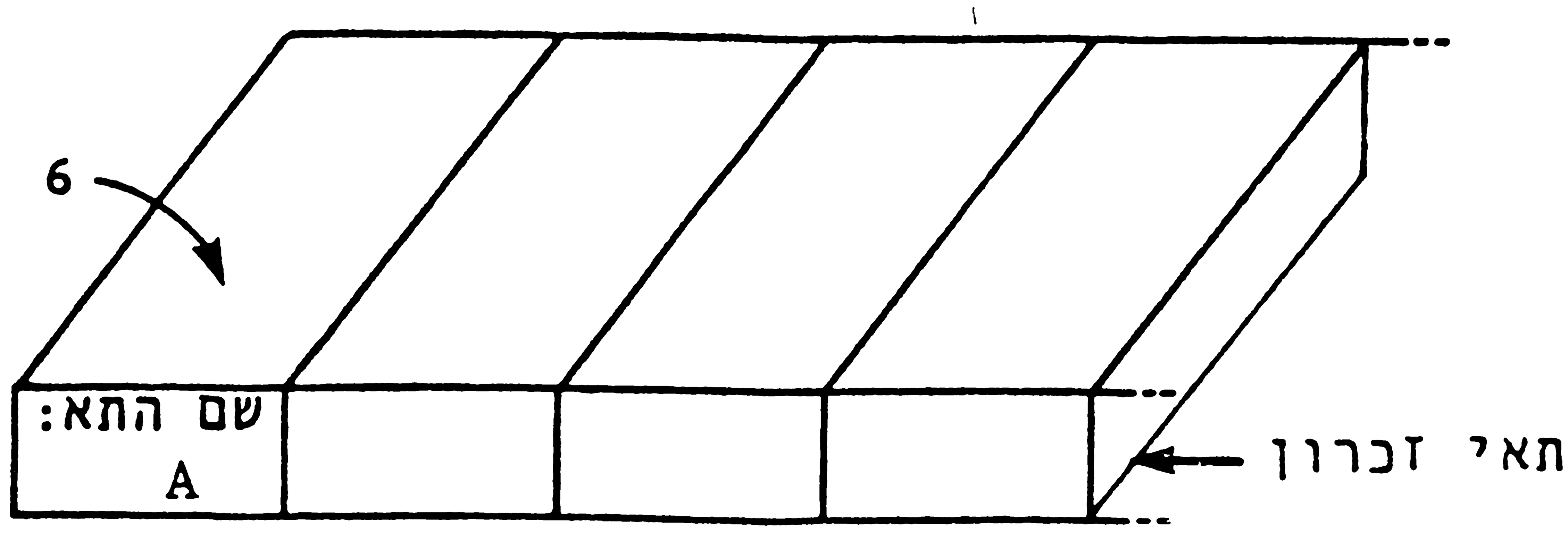
LET A=6

הוא ביצע את הפעולה המעניינת הבאה:

המחשב הציב באחד מתאי

הזכרון שלו את המספר 6:

וקרא לתא זה בשם: A

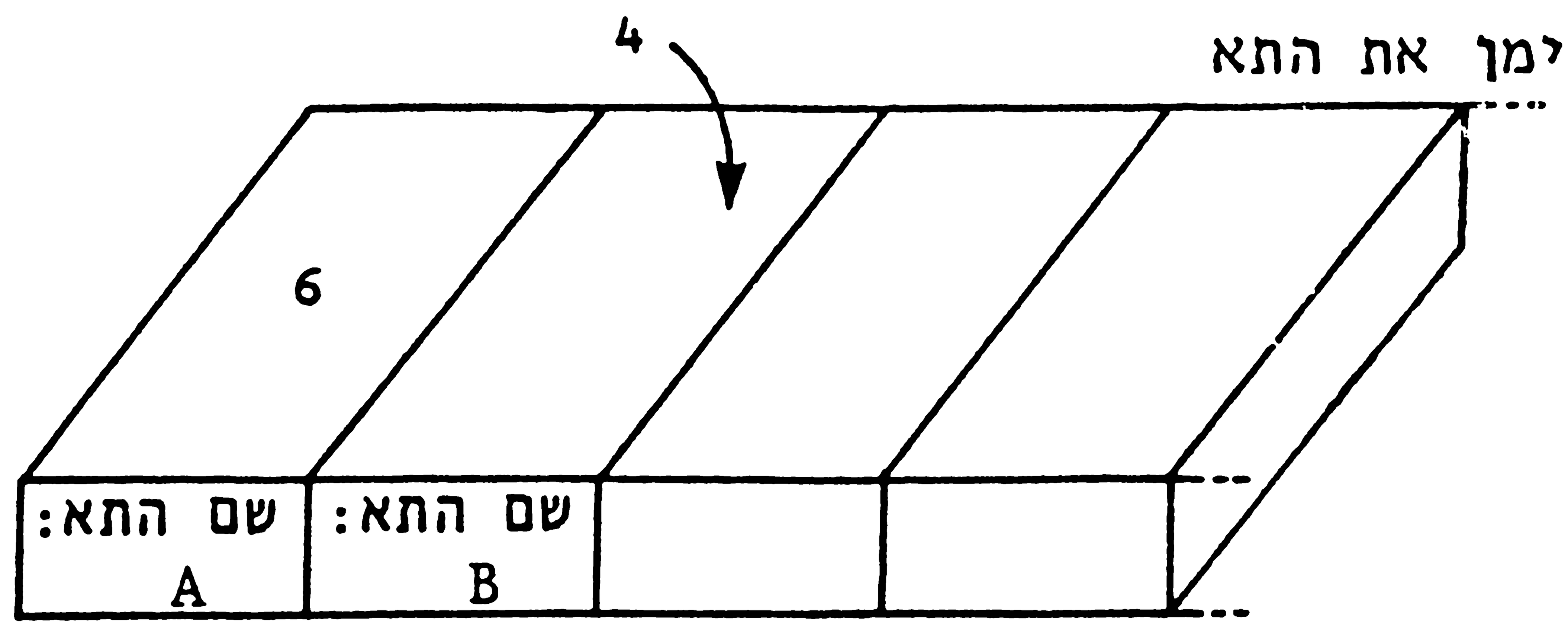


דבר דומה קרה כאשר פקדת עליו LET B=4:

הוא הציב בתא זכרון נוסף

את המספר 4 וסימן את התא

בשם: B



ועכשיו, כאשר פקדת עליו:

PRINT A+B

הוא ביצע את הפעולות הבאות:

- חיפש תא בשם A, מצא אותו, והעתיק את המספר הנמצא בתוכו (6) אל מרכז החישובים הנמצא "במוחו".
- המחשב המשיך וחיפש תא בשם - B, מצא אותו והעתיק את המספר שהיה בו אל אותו מרכז חישובים.
- מרכז חישובים זה נקרא בשפה מקצועית: היחידה האריתמטית. ביחידה האריתמטית המחשב ביצע את החיבור של 6 ו-4 והציג את התוצאה על המסך.

האם לאחר שהמחשב ביצע את פעולת החיבור עדיין נמצאים המספרים 6 ו-4 בתאים A ו-B?
בדוק זאת על המחשב:

פקוד עליו להדפיס את ערכו של A ‏(PRINT A).

ומה לגבי ערכו של B?

מסקנה כדי לבצע פעולות ב-A וב-B העתיק המחשב את ערכם מן התאים שלהם. הערך של A ו-B נשאר בזכרון!

מה יתקבל על המסך אם תפקוד על המחשב:

PRINT A*B

(להזכירך: * - הוא סימן הכפל במחשב).

פקוד על המחשב לבצע את פעולת הכפל ובדוק אם צדקת.

עכשיו פקוד על המחשב:

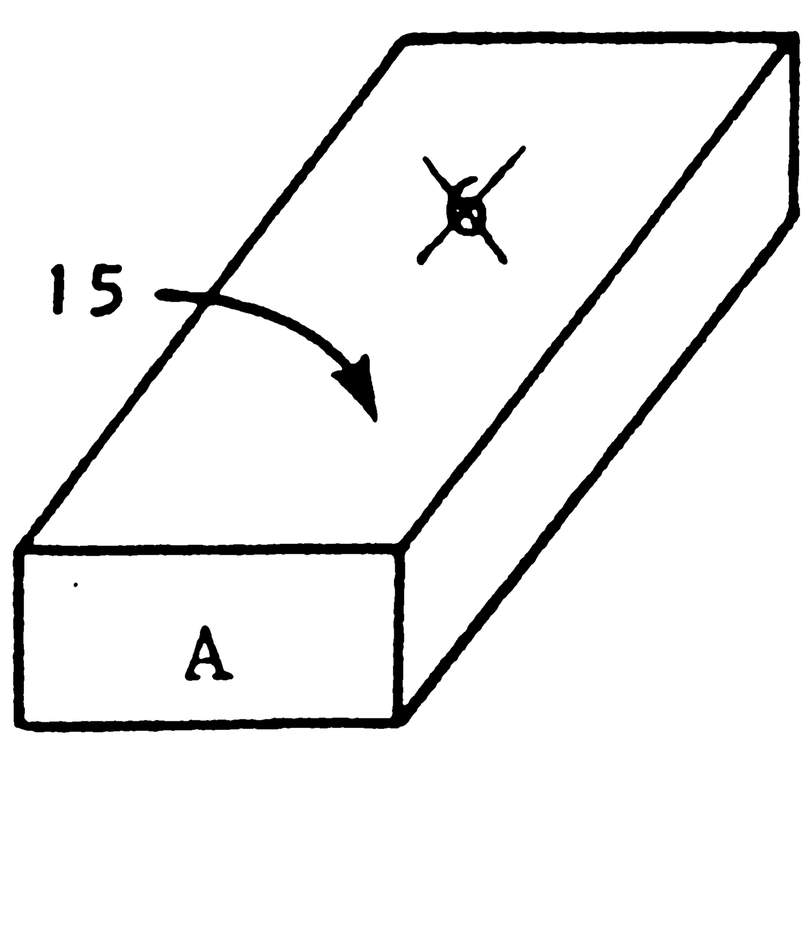
LET A=15

מה ידפיס המחשב לאחר שתפקוד עליו PRINT A: __________
(15,15,6,5 ו-6 ביחד)

פקוד עליו להדפיס את A! האם צדקת?

הסבר

- כאשר פקדת קודם LET A=6, הכניס המחשב לתא זכרון את המספר 6 וקרא לתא - A.
- כאשר פקדת אח"כ LET A=15, חיפש המחשב בזכרונו לראות אם קיים תא בשם A. הוא, כמובן, מצא תא כזה, ואז הכניס פנימה את המספר 15, ומחק את המספר 6:



נסכם:

נוכחת לדעת כי A יכול לייצג מספרים כלשהם בהתאם למה שנמצא בתא הזכרון A.

מסיבה זו, "יצור" כמו A נקרא בשם **משתנה.**

ליתר דיוק - משתנה מספרי. (בהמשך היחידה תכיר משתנה נוסף הנקרא - משתנה מחרוזתי.)

☐ פקוד על המחשב לתת למשתנה B את הערך 5.

? מה ידפיס המחשב כאשר תפקוד עליו:

```
PRINT A/B
PRINT 3*A
```

☐ נסה!

? מה יעשה המחשב כאשר תפקוד עליו:

```
LET C = A + B
```

במקרה זה הגדרנו משתנה חדש, C, אשר שווה לסכום שני המשתנים A ו-B. (בדוק זאת ע"י PRINT C)

? בדוק מה ידפיס המחשב במקרה הבא:

```
PRINT D
```

הסבר

המחשב בודק אם קיים בזכרונו משתנה בשם D. כמובן שהוא לא מצא משתנה בשם זה ולכן D קיבל את הערך Ø.

? מה יקרה כאשר תפקוד:

```
PRINT A/F
```

? מדוע מודיע לך המחשב כי ניסית לחלק באפס?

(תשובה 1 בעמוד 62)

קיצורי־דרך בכתיבה

עד כה הגדרנו משתנים בעזרת פקודת LET.

המחשב שלך יודע לעשות זאת גם אם תוותר על כתיבת המילה LET:

נסה לכתוב: A=5Ø

בקש מן המחשב להדפיס את ערכו של A.

שמות של משתנים

איזה שם מותר לתת למשתנה?

השם שהמחשב נותן לתא בזכרון בו נמצא הערך של המשתנה, חייב להתחיל באות אנגלית. אחרי אות זו יכולות לבוא עוד אותיות אנגליות או ספרות, כרצונך.

אילו מבין שמות המשתנים הבאים אינו חוקי? :

IBM, SUM, B, 2D, A1B2, 25, PRINT

(תשובה 2 בעמוד 62)

כתוב במחשב: 1X=2

(כדי לבדוק אם 1X יכול להיות שם של משתנה).

נסה כעת להדפיס את ערך המשתנה (PRINT 1X).

המחשב הדפיס: 1 Ø.

אמור למחשב להדפיס את התוכנית. (אינך זוכר איך? - פנה ליחידה 1 עמוד 29)

איזו שורה נוספה עכשיו לתוכנית שכבר נמצאת בזכרונו של המחשב?

אתה רואה כי המחשב פשוט התיחס ל-1 כאל מספר בתוכנית, וכשפקדת עליו PRINT 1X הוא הדפיס את המספר 1 ולידו את הערך של המשתנה X (Ø).

מחק את שורה 1 מהתוכנית!

(אינך זוכר איך? - ראה יחידה 1 עמוד 34)

נסכם

- שמו של משתנה חייב להתחיל באות אנגלית.
- אחרי אות זו יכולות לבוא אותיות נוספות או ספרות (כרצונך).

אתגר מחשבתי

הגדר במחשב משתנה חדש, X, שיהיה שווה ל-1Ø:

X = 1Ø

(להזכירך: ניתן לכתוב X=1Ø במקום LET X=1Ø - וכמובן ENTER)

מה יהיה ערכו של X לאחר שתפקוד על המחשב:

LET X = X+1

כדי לדעת אם צדקת, פקוד על המחשב לבצע פקודה זו (אפשר גם בלי LET) ואח"כ פקוד עליו להדפיס את X.

קיבלת כי X שווה ל _____ (1, 1Ø, 11).

כיצד התקבלה תוצאה זו?

הפקודה X=X+1 אין פירושה ש-11=1Ø!

הפקודה X=X+1 פירושה עבור המחשב הוראה לביצוע סדרה של פעולות:

- המחשב מבצע תחילה את מה שכתוב <u>בצד ימין</u> של השוויון: במקרה שלנו לוקח את הערך הנוכחי של המשתנה X ומוסיף לו 1: (1Ø+1).
- את התוצאה המחשב <u>מציב</u> במשתנה ששמו רשום בצד שמאל של השוויון: במקרה שלנו מופיע שם שוב המשתנה X ולכן הוא מקבל עכשיו את הערך 11.

משימה

ועכשיו, לאחר שהבנת את מושג המשתנה, תוכל לפצח את המשימה שהטלנו עליך קודם לכן.

הצג את התוכנית, שכבר נמצאת בזכרונו של המחשב, על המסך.

(זאת התוכנית שכבר כתבת בעמוד 6)

עליך לשנות את התוכנית כך שהמחשב ידפיס במרכז המסך מספרים מ-1 והלאה. כלומר: ידפיס 1 במרכז המסך, מיד לאחר מכן ידפיס באותו מקום 2 (ה-1 נמחק, כמובן), עליו 3 וכן הלאה.

בידך כל הכלים התכנותיים לכתוב את התוכנית - נסה בכוחות עצמך.

מתקשה?

רמז 1: עליך להוסיף עוד שתי שורות לתוכנית.
(ואולי רמז זה יספיק לך?)

רמז 2: השלם את החסר:

```
4Ø ___ = ___ + 1
5Ø GO TO ___
```

אגב האם המספרים רצים מהר מדי על המסך?

- עצור את ריצת התוכנית (אינך זוכר איך? - ראה יחידה 1 עמוד 39).
- הכנס "פסק-זמן" במקום המתאים בתוכנית, בדומה לזה שהכנסת כדי להאט את קצב ההתעמלות של אי-ציק ביחידה 1 עמוד 57.

(תשובה 3 בעמוד 62)

התבונן במסך: האם זה מזכיר לך התחלפות מספרים בשעון דיגיטלי? - בפרק ג תבנה בעצמך שעון כזה.

נלך צעד צעד ונראה כיצד התוכנית עובדת:

שורה 2Ø: אחרי ביצוע שורה זאת המשתנה N שווה ל:___(3,2,1)

שורה 3Ø: המחשב מדפיס את ערכו של N.

שורה 4Ø: N מקבל ערך <u>חדש</u> השווה לערך <u>הקודם</u> בתוספת 1:

```
N = N + 1
↑   ↑
חדש קודם
```

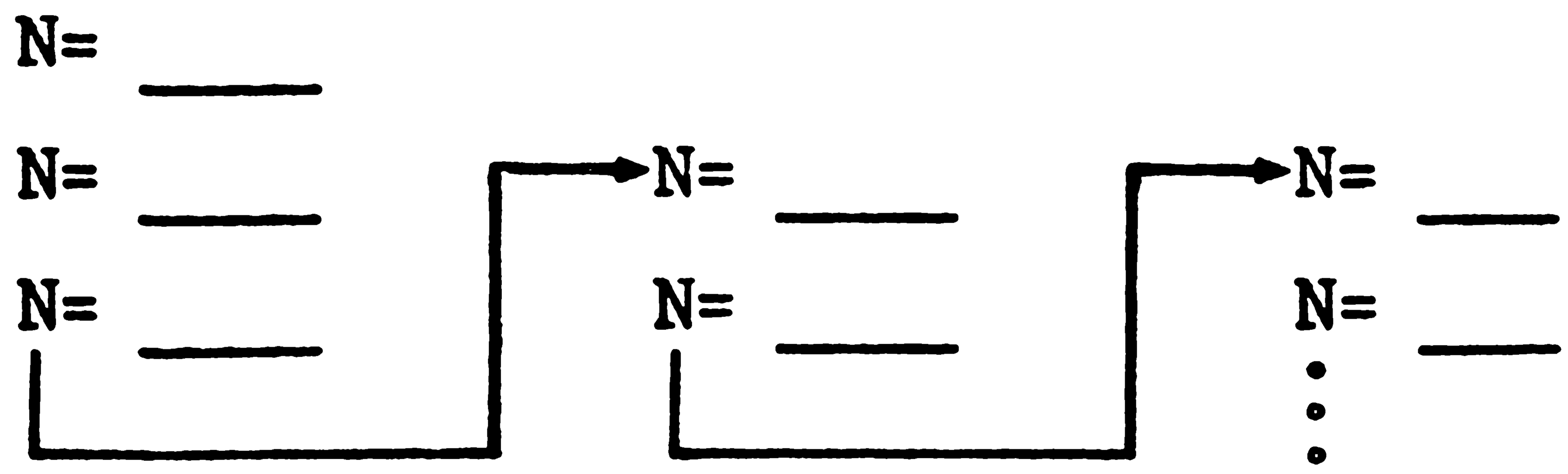
אחרי ביצוע שורה זו בפעם הראשונה, N מקבל את
הערך: ___ (1, 2, 3)

שורה 5Ø: פקודה זו כבר מוכרת לך מיחידה 1. הפקודה מחזירה
את המחשב לשורה ___ בתוכנית.
וכך הלאה...

הייה אתה המחשב

כדי לעקוב אחר הגלגולים שעובר המשתנה N בזמן ריצת התוכנית,
הייה אתה המחשב - עבור שורה שורה וכתוב את ערכו של המשתנה
N בכל שורה בתוכנית הבאה:

```
2Ø N=1
3Ø PRINT N
4Ø N=N+1
5Ø GO TO 3Ø
```



כדי לבדוק את תשובותיך השווה לתשובה 4 בעמוד 62.

התוכנית שבמחשב עדיין רצה על המסך שלך?
אם לא תעצור את המחשב - האם הוא ימשיך לספור עד אינסוף?
כמעט! המחשב יכול לספור עד המספר העצום 10^{38}!
(הספרה 1 עם "זנב" של 38 אפסים)

משימות

בצע בתוכנית שבמחשב את השינויים הבאים:

- שנה את התוכנית כך שיודפסו רק המספרים האי-זוגיים:
  1, 3, 5, 7 - - -

(תשובה 5 בעמוד 62)

- בצע שינוי כזה שיודפסו המספרים: 5, 1Ø, 15, - - -

מה יקרה אם תשנה את שורה 5Ø ל:

```
5Ø GO TO 2Ø
```

פינת המשחק: "בול 1000"

בעזרת מה שלמדת זה עתה תוכל לשחק משחק קטן עם עצמך או עם חבריך:

כתוב את התוכנית הבאה:

```
5  CLS
10 LOCATE 10, 20 : PRINT "BOOM"
20 B=B+1
30 GO TO 20
```

הרץ את התוכנית.

האם המחשב סופר על המסך? _____ (כן/לא)

דע לך כי מוחו של המחשב סופר ללא הרף במהירות רבה!

(המחשב מתרוצץ ללא הפסקה בין שורה 20 לבין שורה 30)

רוצה להשתכנע שהוא סופר?

עצור את ריצתו ופקוד עליו להדפיס את ערכו של המשתנה B.

(אינך זוכר כיצד? ראה עמוד 7)

המספר שהודפס הוא ערכו של B כשהתוכנית נעצרה.

חזור והרץ את התוכנית. עצור את המחשב והצג שוב את ערכו של B. מה ערכו של B הפעם?

וכעת למשחק עצמו:

מטרת המשחק: לעצור את המחשב קרוב ככל האפשר למספר 1000.

שיטת המשחק: כל אחד בתורו מקבל 5 נסיונות. בכל ניסיון הוא מריץ את התוכנית ועוצר אותה כאשר הוא <u>חושב</u> ש-B הגיע ל-1000. כל אחד זוכר את השיא שלו. (המספר הקרוב ביותר ל-1000).

מי שהגיע למספר הקרוב ביותר ל-1000 - ניצח!

סיימת לשחק?

מחק את התוכנית מהזכרון ועבור הלאה...

חידה

מה יקרה כאשר תריץ את התוכנית הבאה?

```
10 P=P+100
20 SOUND P, 1
30 GO TO 10
```

לאחר שחשבת...

☐ הרץ את התוכנית עד שהיא תפסק מעצמה.

אם פעלת כראוי המחשב הפסיק לרוץ והודיע:

OVERFLOW IN 20

מדוע הוא הודיע זאת?

☐ פקוד על המחשב להדפיס את ערכו של P.

הסבר

אתה כבר יודע, כי המספר הראשון בפקודת ה-SOUND אחראי לגובה הצליל שהמחשב משמיע (ראה יחידה 1 עמוד 43). מספר זה יכול לקבל ערך מכסימלי של 32767, שהוא כמובן הצליל הגבוה ביותר. מאחר שהמשתנה P קיבל את הערך 32800, שהוא גדול יותר ממספר זה, המחשב לא היה מסוגל לבצע את פקודת ה-SOUND - והוא אכן הודיע לך על כך.

? ומה יקרה לדעתך במקרה הבא?

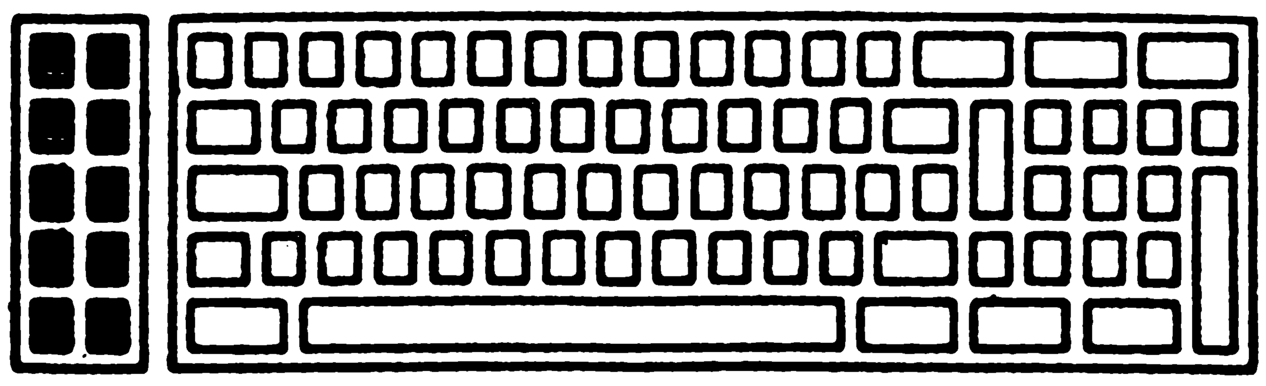
```
5  P=5000
10 SOUND P, 2
20 P=P-100
30 GO TO 10
```

? מה לדעתך יהיה הערך של P כאשר המחשב יפסיק לבצע את התוכנית?

☐ בדוק זאת במחשב.

מקשים מתוכנתים

את סיום הפרק נקדיש ללימוד נוסף של המקשים המתוכנתים - אלו הם 10 המקשים הנמצאים בצדו השמאלי של לוח המקשים:



פקוד WIDTH 8Ø ורשימה מלאה של 10 המקשים מופיעה בתחתית המסך.

משנים את תפקידם של מקשי הפקודות:

עד כה, כאשר רצית לכתוב את הפקודה SOUND, נאלצת לתקתק את כל הפקודה אות אחר אות. הרבה יותר נוח יהיה, אם בלחיצה על מקש אחד היתה נכתבת כל הפקודה בדיוק כמו אותן פקודות המופיעות בתחתית המסך.

פקודת KEY

בעזרת הפקודה KEY (קי), שפירושה מקש, תוכל לשנות את תפקידו של אחד ממקשי הפקודות F כך, שבלחיצה עליו המחשב ידפיס באופן אוטומטי את הפקודה SOUND. בתחתית המסך מופיע במלבן השישי "LPT 1:", בשלב זה היא אינה שימושית ואנחנו נחליף אותה בפקודה SOUND.

פקוד על המחשב לבצע:

```
KEY 6, "SOUND"
```

אגב, תוכל להשתמש במקש F9 כדי להדפיס את הפקודה KEY.

לחץ ENTER - מה מופיע עכשיו במלבן השישי?

אם פעלת היטב, אז פקודת ה-SOUND הופיעה במלבן ה-6 שבתחתית המסך

עכשיו לחץ על מקש F6.

המחשב הדפיס באופן אוטומטי את הפקודה SOUND.

באותו אופן תוכל לשנות את כל אחד מ-10 מקשי הפקודות לפי בוחותך

האם לאחר שתפקוד NEW יחזור מקש F6 לקדמותו?

בסה!

האם לאחר שתכבה את המחשב יחזור מקש F6 לקדמותו? (או שתלחץ על שלושת המקשים Alt+Ctrl ו-Del).

נסה!

שנה את מקש 7 כך שבלחיצה עליו המחשב ידפיס את הפקודה SOUND.

הערה: רצוי שתוסיף רווח בסוף הפקודה: "SOUND "
וכך לא תשכח להדפיס אותו. ↑ רווח

שנה את אחד המקשים כך, שבלחיצה עליו המחשב ידפיס באופן אוטומטי את שמך!

שים לב! המחשב מסוגל לקלוט מכסימום 15 סימנים (כולל רווחים), כך שאם שמך ארוך מ-15 סימנים המחשב ידפיס את שמך רק באופן חלקי.

KEY OFF

פקוד עכשיו KEY OFF.

אם פעלת היטב המחשב כיבה את רשימת המקשים המתוכנתים שבתחתית המסך ונוספה שורה נוספת (ה-25) למסך שתוכל להדפיס בה במקרה הצורך.

האם מקשי הפקודות עצמם פועלים?

נסה!

KEY ON

אנו מתארים לעצמנו כי אתה כבר יודע מה יקרה כאשר תפקוד:

KEY ON

נסה!

לסיום הפרק...

בפרק זה למדת את אחד המושגים החשובים ביותר בתכנות בשפת ה-BASIC: את מושג המשתנה. למעשה, בכל תוכנית שתכתוב מכאן ואילך יופיעו משתנים.

פרק ב
לולאות סופיות

עכשיו, לאחר שהכרת את מושג המשתנה, נוכל לפנות ולפצח את הפקודות:

```
30 FOR N=0 TO 7
°
°
°
70 NEXT N
```

שהופיעו בתוכנית הצלילים הצבעונית בפתיחת היחידה, בעמוד 5.

לולאות FOR-NEXT

האם אתה זוכר את התוכנית:

```
10 PRINT "שמך"
20 GO TO 10
```

מיחידה 1?

הפקודה GO TO המופיעה בשורה 20 גרמה לכך שהמחשב הסתובב בלולאה אינסופית וביצע פעם אחר פעם את שורה 10. פעולה זו נפסקה רק כאשר עצרת את התוכנית.

בעזרת ההוראות FOR-NEXT תוכל להכתיב למחשב כמה פעמים לבצע כל הוראה:

האם תוכל לנחש כמה פעמים ירשום המחשב את שמך בתוכנית הבאה?

```
10 FOR N=1 TO 5   ← אל תשכח רווח אחרי כל מילה.
20 PRINT "שמך"
30 NEXT N
```

העתק את התוכנית למחשב והרץ אותה!

(לפני כן נקה את הזכרון מכל תוכנית קודמת וכן את המסך.)

אתה רואה כי שמך הופיע 5 פעמים זה מתחת לזה.

שנה את התוכנית כך, שהמחשב ישכפל את שמך <u>8</u> פעמים.
(תשובה 6 בעמוד 62)

נסכם:

כאשר אתה כותב תוכנית עם צמד הפקודות <u>FOR-NEXT</u> אתה יכול לשלוט במספר הפעמים שהמחשב מבצע את השורות הכתובות בין צמד פקודות זה.

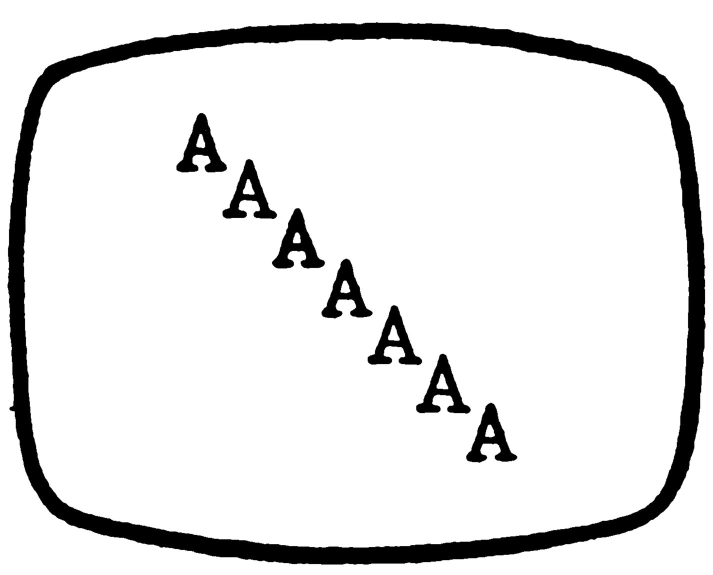
למשל:

```
10 FOR N=1 TO 20
┌20 .............
│30 .............
└40 .............
50 NEXT N
```

כמה פעמים יבצע המחשב את שורות 2Ø, 3Ø ו-4Ø?
(תשובה 7 בעמוד 62)

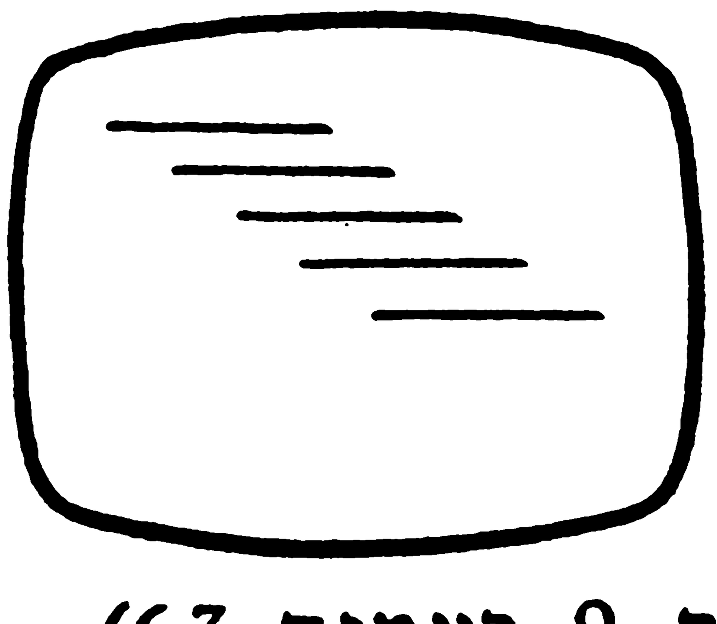
משימה

א. כתוב תוכנית שתרשום שבעה A לאורך אלכסון במרכז המסך:
(השתמש ב-FOR-NEXT)



(תשובה 8 בעמוד 63)

ב. כתוב תוכנית שתרשום את שמך 5 פעמים בצורה כזאת על המסך:
(שוב, השתמש ב-FOR-NEXT)



(תשובה 9 בעמוד 63)

משתנה לולאה

מחק את התוכנית ועבור הלאה:

מה תבצע התוכנית הזאת:

```
5  X=1
1Ø FOR N=1 TO 2Ø
2Ø PRINT X
3Ø X=X+1
4Ø NEXT N
5Ø PRINT "SOF" (SOF = סוף)
```

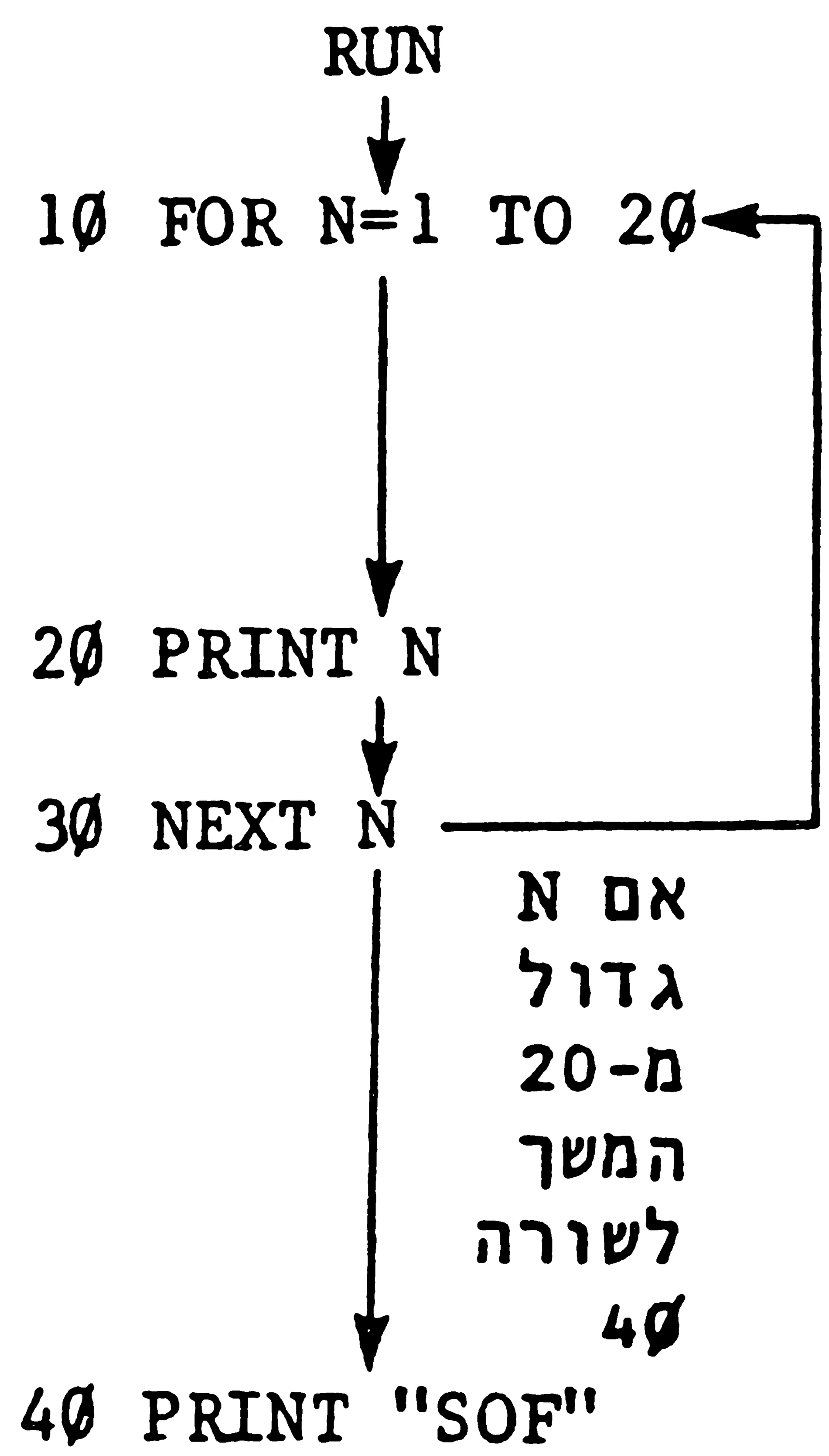
לאחר שחשבת - העתק את התוכנית והרץ אותה.

למען האמת ניתן לקצר את התוכנית שכתבנו זה עתה ולכתוב אותה כך:

```
1Ø FOR N=1 TO 2Ø
2Ø PRINT N
3Ø NEXT N
4Ø PRINT "SOF"
```

<u>שים לב לדבר מעניין:</u>

המשתנה N המופיע בשורה 2Ø, מופיע גם בשורה 1Ø ובשורה 3Ø!
כדי לעזור לך להבין את אשר קורה למשתנה N נעקוב אחר ביצוע התוכנית שורה אחר שורה:



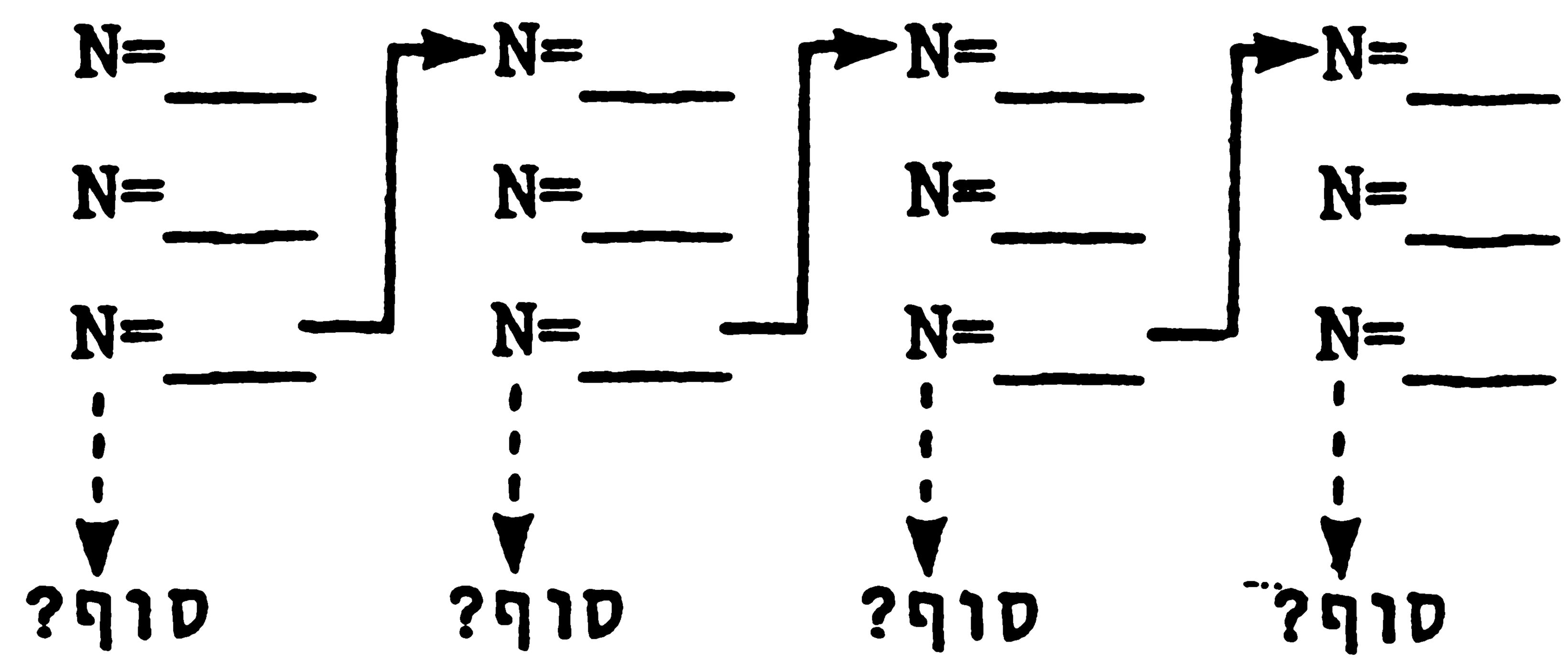
בשורה זו נאמר למחשב: תן למשתנה N להתחיל מ-1 (FOR N=1) ואל תרשה לו לעבור את 2Ø (TO 2Ø)

כאן נאמר למחשב להדפיס את ערכו של N

המחשב מקבל פקודה להגדיל את המשתנה N ב-1 ולחזור לשורה 1Ø. <u>אבל</u> - אם N כבר עבר את 2Ø והגיע ל-21 - המחשב לא יחזור לשורה 1Ø אלא ימשיך הלאה - לשורה 4Ø!

בדוק עצמך אם הבנת: הייה אתה המחשב וכתוב את ערכי המשתנה N במקומות הריקים:

```
RUN
 ↓
10 FOR N=1 TO 3 ---
20 PRINT N --------
30 NEXT N  --------
40 PRINT "SOF"
```



החלט גם, מתי אתה כמחשב מדפיס "סוף".

(תשובה 10 בעמוד 63)

נחזור לתוכנית הנמצאת בזכרונו של המחשב:

מה יקרה אם נשנה את שורה 10 ל:

```
10 FOR N=5 TO 20
```

התוכנית תספור מ___ עד ___.

נסה וראה אם צדקת.

מה יקרה אם תמחק לגמרי את שורה 30?

מחק את שורה 30 מן התוכנית והרץ אותה.

המחשב מודיע:

```
FOR WITHOUT NEXT IN 10
```

המחשב סירב לבצע את שורה 10 מאחר שאין בתוכנית הוראת NEXT שתחזיר את המחשב בחזרה לשורה 10.

מחק את התוכנית מן הזכרון.

מה תעשה התוכנית הבאה:

```
10 FOR N=0 TO 5
20 PRINT N*N
30 NEXT N
```

חשוב לפני שתבדוק עצמך על המחשב!

משימה

זוכר את המשימה אשר בה ביקשנו ממך לשרטט שבע פעמים A באלכסון? (ראה עמוד 19)

עכשיו, כאשר אתה יודע על קיומו של משתנה הלולאה, תוכל לקצר תוכנית זאת. נסה!

(תשובה 11 בעמוד 63)

לולאת השהייה

מה קורה כאשר אין כלל שורות בין צמד הפקודות FOR-NEXT היוצרות את הלולאה:

```
10 PRINT "START"        (START באנגלית - התחלה)
20 FOR T=1 TO 1000
30 NEXT T
40 PRINT "END"          (END באנגלית - סוף)
```

מה עושה תוכנית זו? מה תפקידן של שורות 20 ו-30?

לאחר שחשבת...

נקה את הזכרון והכנס את התוכנית הזאת.

מה עליך לעשות כדי להגדיל פי שניים את הזמן בין הופעת ה-START על המסך לבין ה-END?

(תשובה 12 בעמוד 63)

צמד השורות 20 ו-30 יצר השהייה.

כאשר המחשב מגיע לשורות 20 ו-30 הוא נמצא בלולאה, והוא מבצע אותה ___ (10, 100, 1000) פעמים. (המשתנה N מתחיל מ-1 ורק כאשר N גדול מ-1000 המחשב יוצא מן הלולאה.)

הזמן שבו ביצע המחשב 1000 סיבובים בלולאה יוצר השהייה. מובן, לכן, שככל שהמחשב מבצע יותר סיבובים בלולאה - ההשהייה ארוכה יותר.

למען האמת זאת השיטה המקובלת כדי להשיג "פסק-זמן". כאשר כותבים את פקודות ה-FOR-NEXT כדי להשיג פסק זמן כדאי לכתוב אותן בזו אחר זו באותה שורה.

באופן הבא:

```
20 FOR T=0 TO 1000 : NEXT T
   \______________/   \____/
          1              2
```

כאשר המחשב גומר את פקודה 1 בשורה הוא עובר לבצע את פקודה 2 ולאחר מכן חוזר לפקודה 1 וכך חוזר 1000 פעמים!
רק כאשר 1000 < T ‏(T גדול מ-1000) המחשב ימשיך לשורה הבאה.

שנה את התוכנית כך ששורה 30 תוכנס לתוך שורה 20.
הרץ את התוכנית וראה כי מתקבלת אותה תוצאה.
(האם זכרת למחוק את שורה 30?)

התוכנית העייפה

התבונן בתוכנית הבאה:

```
 5  CLS
10  PRINT X;
20  FOR T=1 TO X * X : NEXT T
                     ↑
30  X=X+1           כפול
40  GO TO 10
```

מדוע קראנו לתוכנית זו "התוכנית העייפה"?
הרץ את התוכנית ובדוק בעצמך.
(אינך מבין? פנה לתשובה 13 בעמוד 63)

חגיגת צבעים

בנה תוכנית אשר תצבע את כל המסך בצבעים משחור 0 ועד צבע לבן (15).
הצבעים מתחלפים מהר מדי?
הוסף שורה שתיצור השהיה.
נגמרו הצבעים? - המחשב מתחיל מן ההתחלה.

(תשובה 14 בעמוד 64)

עכשיו שנה שורה בתוכנית כך, שהמחשב יצבע את השוליים בצבעים היורדים מ-15 עד ל-0 בזמן שצבעי המסך עולים מ-0 עד 15.

(תשובה 15 בעמוד 64).

? מה עליך לעשות כדי שהמחשב ישמיע צליל שגובהו הולך וגדל?
(תשובה 16 בעמוד 64)

? האם התוצאה המתקבלת על המסך מזכירה לך משהו?
ובכן, דע לך כי למעשה פיתחת תוכנית הדומה לזו שהופיעה בתחילת היחידה בעמוד 5.

נסכם:

- השורה 5Ø:

```
┌→1Ø .....
│ :
└─5Ø GO TO 1Ø
```

יוצרת **לולאה אינסופית** המחשב חוזר ומבצע את הפקודות בין השורות 1Ø ו-5Ø פעם אחר פעם ללא הגבלה.

- השורות 1Ø ו-5Ø:

```
┌→1Ø FOR N=1 TO 1Ø
│ :
└─5Ø NEXT N
```

יוצרות **לולאה סופית** המחשב חוזר ומבצע את הפקודות כמספר הפעמים שרשום בשורה 1Ø.

- המשתנה N בלולאה הסופית נקרא **משתנה לולאה.**

בהמשך יחידת הלימוד נעשה שימוש נרחב במושגים אלה!

פרק ג
בנית שעון דיגיטלי

מבוא

עכשיו, לאחר שהבנת את פקודות הלולאה FOR-NEXT, אתה יכול לבנות במחשב שלך שעון דיגיטלי.
אנו מתארים לעצמנו כי מצוי בביתך שעון דיגיטלי כלשהו.
רצוי מאוד ששעון כזה יהיה לידך בזמן שאתה בונה שעון דיגיטלי במחשב.

ספירת השניות

תחילה נתרכז בבניית הצגת השניות של השעון הדיגיטלי:

התבונן בשעון הדיגיטלי שלידך,על הספרות המציגות את השניות,וענה:
השעון מתחיל לספור את השניות מ- ____ (Ø, 1).
האם מופיע גם המספר 6Ø בסוף הדקה? ___ (כן/לא).

אתה רואה כי השעון מתחיל מ-Ø ומגיע עד 59. ואז במקום לרשום 6Ø הוא חוזר ורושם Ø ומתחיל מן ההתחלה.
כדי לגרום למחשב להדפיס שניות על המסך נבנה את התוכנית בשלבים:

1 כתוב תוכנית שתדפיס, במרכז המסך בערך, מספרים המתחלפים מ-Ø עד 59. (כתוב את התוכנית ברוחב מסך 4Ø).

המספרים המופיעים על המסך מתחלפים במהירות גדולה מאוד.

☐ הכנס לולאת השהייה לתוך התוכנית.

(תשובה 17 בעמוד 64)

2 בשלב שני הוסף פקודה לתוכנית שתגרום למחשב לחזור ולספור מ-Ø כל פעם שהוא מגיע ל-59.

(תשובה 18 בעמוד 64)

הכנסת השעון לקצב שניות אמיתי

האם קצב הופעת המספרים על המסך ____ (מהיר יותר, איטי יותר) מזה של השעון הדיגיטלי?

איזו השהייה עליך להכניס על מנת להתאימו לקצב של שניות?

```
FOR T=1 TO _?_ : NEXT T
```

בצע כמה ניסויים בעזרת שעון דיגיטלי עד אשר תגיע לקצב הקרוב ביותר לזה של השניות בשעון הדיגיטלי.

(תשובה 19 בעמוד 64)

נפנה לספירת הדקות

התבונן בשעון דיגיטלי: מה קורה כאשר השעון הדיגיטלי משלים לספור דקה שלמה?

במקרה זה סיפרת הדקות קופצת ב-1.

עליך להוסיף לתוכנית פקודות מתאימות כך, שהשעון יספור גם דקות.

(רמז: עליך להוסיף לולאת FOR-NEXT מתאימה לתוכנית.)

הערה: ברגע הפעלת השעון, דאג לכך שהמסך יראה כך:

```
Ø : Ø
↑   ↑
```

שניות דקות

(תשובה 20 בעמוד 64)

שתילת לולאות אחת בתוך השניה

בוא נעצור לרגע ונתבונן בתוכנית שלנו:

בתוכנית ישנן 3 לולאות FOR-NEXT:

```
FOR D=O ....
FOR S=O ....
FOR T=TO ___ : NEXT T
NEXT S
NEXT D
```

לולאת השהייה | לולאת שניות | לולאת דקות

אתה רואה כי ניתן להשתמש במספר רב של לולאות בתוכנית "השתולות" האחת בתוך השניה.

האם המחשב מסוגל לבצע תוכנית שבה יש לולאות FOR-NEXT מצטלבות כמו זו:

```
FOR S= ....
FOR N= ....
NEXT S
NEXT N
```

(תשובה 21 בעמוד 64)

משימה

אם עדיין לא במאס לך השעון - הוסף לו גם שעות!

(ואולי גם תאריך?)

קיצור כתיבה של המחשב

עד כה נהגנו לכתוב את שם המשתנה אחרי NEXT:

לדוגמא: NEXT S

למען האמת, המחשב מוכן לקבל את פקודת ה-NEXT גם ללא שם המשתנה:

```
FOR T=1 TO 785 : NEXT
```

האם אין סכנה שהמחשב יתבלבל, כאשר עליו לרוץ בתוך לולאות:

```
FOR D =
FOR S =
:
NEXT → NEXT של מה?
NEXT → NEXT של מה?
```

אל תדאג!

כבר אמרנו שאסור לבנות לולאות מצטלבות. ולכן, גם אם אינך כותב את שם המשתנה בפקודת NEXT, המחשב יודע למה התכוונת:

```
FOR D =
FOR S =
:
NEXT
NEXT
```

המחשב משדך באופן אוטומטי את ה-FOR וה-NEXT הפנימיים ביותר וכך הלאה...

המלצה: אל תתעצל וכתוב את שם המשתנה בפקודת NEXT במקומות בהם <u>אתה</u> עלול להתבלבל.

גם למחשב יש שעון דיגיטלי

האם, כשהדלקת את המחשב הוא שאל אותך מה התאריך ומה השעה?

אם תשובתך שלילית - סימן שאינך משתמש ב-DOS של המחשב.

יחידה 4 מוקדשת כולה ללימוד נושא זה, ואנו ממליצים כי כבר בשלב זה תקפוץ ותלמד לפחות את הפרק הראשון ביחידה 4.

אם תשובתך חיובית - דע לך כי המחשב יאפשר לך לקבל שעון דיגיטלי מדויק וזאת בעזרת ההוראה: TIME$

כתוב את התוכנית הבאה:

```
10 CLS
20 LOCATE 10, 15 : PRINT TIME$
30 GO TO 20
```

במרכז המסך מתקבל שעון דיגיטלי המציין את הזמן שחלף מהרגע בו הדלקת את המחשב.

☐ עצור את ריצת המחשב.

? האם השעון הפנימי ממשיך לעבוד גם כשהתוכנית בעצרה?

☐ כדי לבדוק זאת התבונן בספרות המוצגות על המסך ותוך כדי כך לחץ על מקש F5 כדי להריץ התוכנית מהנקודה בו היא נעצרה (CONT).

אנו מקווים כי הגעת למסקנה, שהשעון הפנימי עובד כל הזמן!

מאפסים את השעון

☐ הוסף את השורה `5 TIME$ = "00:00:00"`

☐ והרץ את התוכנית.

אתה רואה כי שורה 5 מאפסת את השעון.

? כיצד תכוון את השעון הפנימי לשעה האמיתית?

פרק ד
שטיחים לפי הזמנה

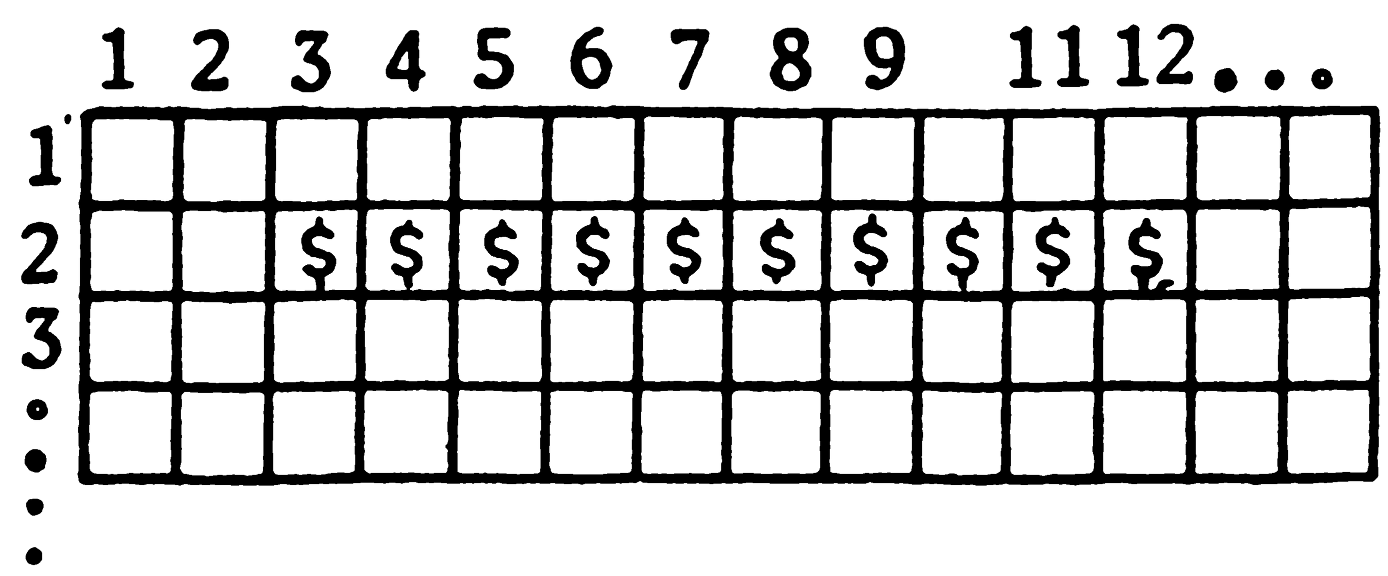
משימות שכפול

נתחיל את הפרק בכמה משימות שכפול:

כתוב תוכנית שתשכפל את הסימן $ 10 פעמים.

במשבצות הבאות:

(רשת מלאה של המסך נמצאת ביחידה 1 עמוד 70)



בתוכנית לא צריך להופיע

יותר מסימן $ אחד.

זקוק לרמז?

השלם את התוכנית הבאה:

```
5  CLS
1Ø FOR X= ___ TO ___
2Ø LOCATE 2, ___ : PRINT "$"
3Ø NEXT ______
```

(תשובה 22 בעמוד 64)

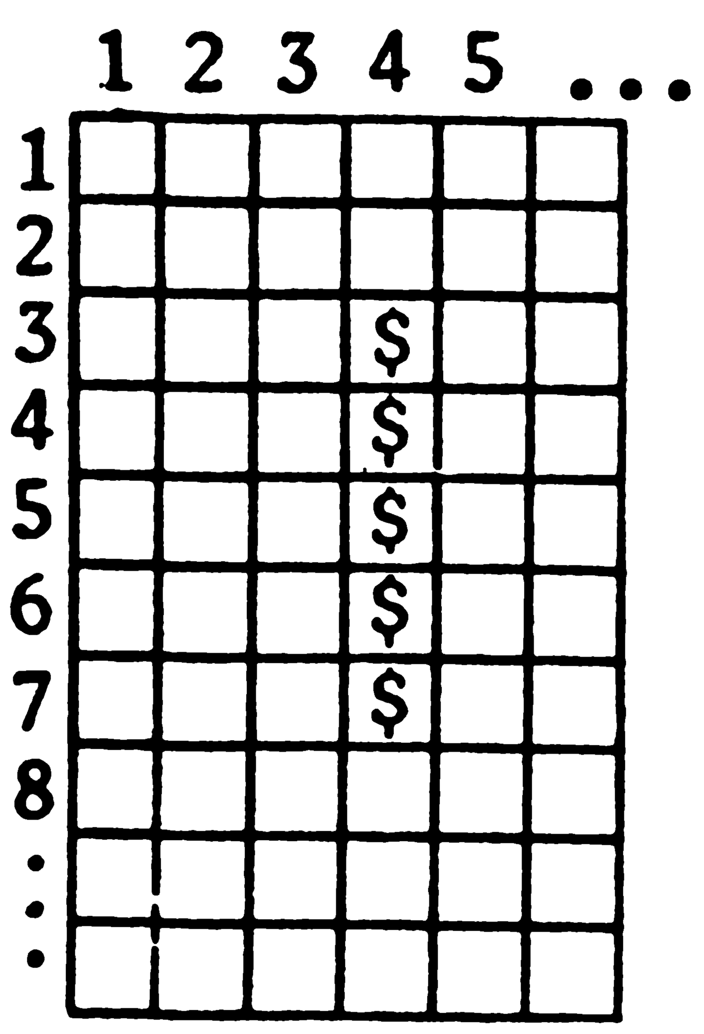
אם התוכנית רצה מהר מדי אולי כדאי שתכניס לולאת השהייה.

הרץ את התוכנית.

כתוב תוכנית דומה שתשכפל את

עמודת הדולרים הבאה:

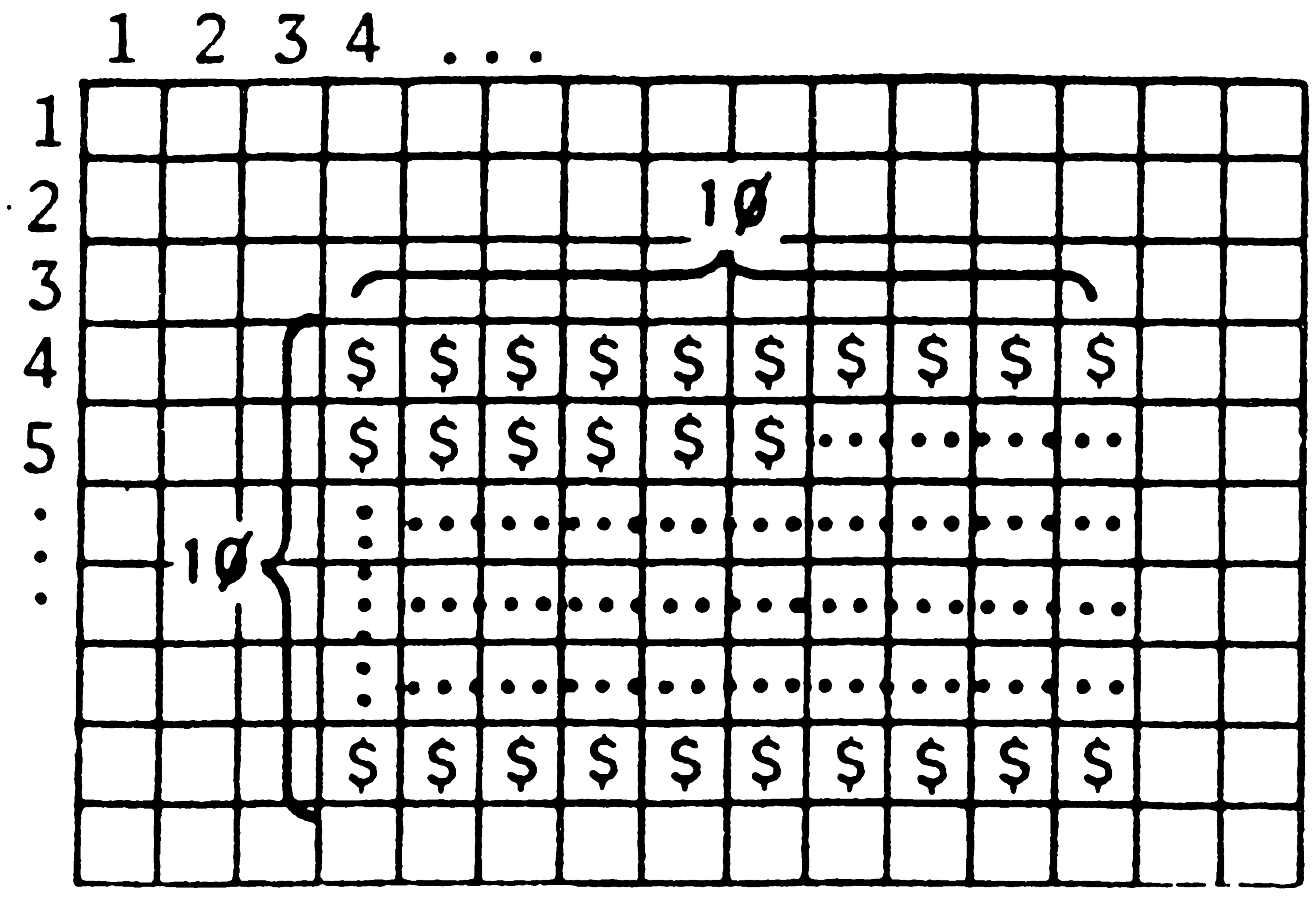
(קרא למשתנה שלך: Y)



(תשובה 23 בעמוד 65)

שטיח של דולרים:

עכשיו עליך לכתוב תוכנית שתיצור "שטיח" (ריבוע) של דולרים:



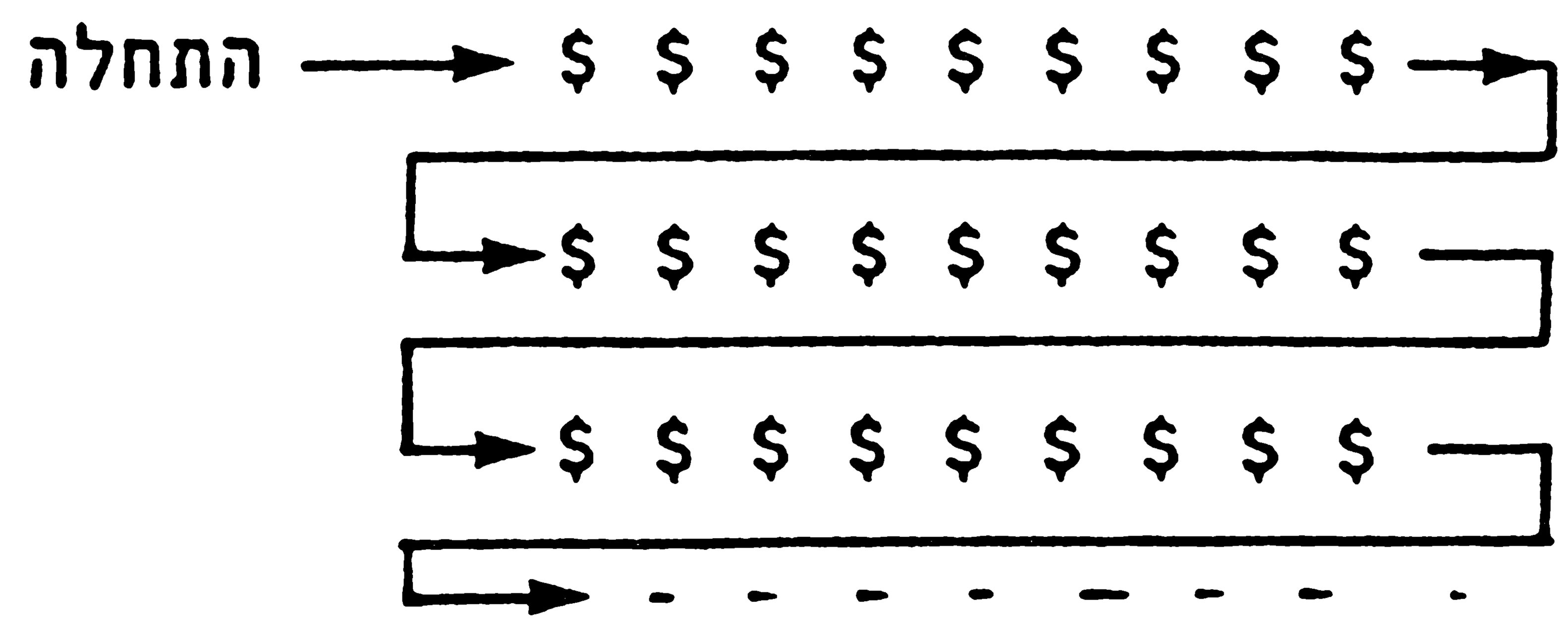
זכור:

- בתוכנית מותר רק ל-$ אחד להופיע!

רמז 1:

אנו רוצים לכתוב תוכנית, אשר תחילה תשכפל את השורה העליונה של הדולרים, ולאחר מכן תעבור לשורה השנייה ותתחיל לרשום אותה מהתחלה. וכך הלאה עד שהיא תשלים ותכתוב את מספר השורות המבוקש.

האיור הבא מדגים כיצד התוכנית מכפילה את הדולרים בשלוש השורות הראשונות:



כיצד תכתוב את התוכנית כך, שכל פעם שהיא משלימה שורה היא תקפוץ, באופן אוטומטי, לשורה הבאה ותתחיל שוב לשכפל דולרים? אם אתה יודע לענות על שאלה זו אולי אתה יכול לכתוב את התוכנית? נסה <u>עכשיו</u> בלי להמשיך לרמז 2.

רמז 2:

בתוכנית הבאה חסרות שתי פקודות:

```
10 CLS
20 ?
30 FOR X=3 TO 12
40 LOCATE Y, X : PRINT "$"
50 NEXT X
60 ?
```

אם תשלים אותן נכון תקבל את התוכנית המבוקשת.

(תשובה 24 בעמוד 65)

☐ הכנס בתוכנית שינויים כאלה, שהשטיח יהיה בגודל 5×5.

☐ שנה את התוכנית כך, שיודפס שטיח של 15 שורות ו-15 עמודות.

קרוב לודאי שאתה נאלץ לבצע שינויים <u>בשני</u> מקומות בתוכנית, בכל פעם שאתה רוצה לשנות את גודל הריבוע:

משימה

☐ שנה את התוכנית כך, שכדי לשנות את גודל השטיח, תצטרך לשנות מספר רק במקום אחד בתוכנית!

(רמז: אתה צריך להשתמש במשתנה נוסף)

התגבר על הבגים לפני שתפנה לתשובה 25 בעמוד 65.

משכללים את התוכנית

אנו מעוניינים שהמחשב ישאל אותך <u>לפני</u> ציור השטיח באיזה גודל שטיח אתה מעונין.

לשם כך עליך ללמוד פקודה חדשה:

INPUT (אִינְפּוּט)-הַכנס לתוכנית:

☐ הוסף את שורה 12:

```
        רווח
          ↓
12 INPUT L
```

ושנה, בשורות 2Ø ו-3Ø את שם המשתנה שקבע את גודל השטיח ל-L אם הוא לא כזה.

☐ הרץ את התוכנית.

על המסך מופיע סימן שאלה: ?

עכשיו המחשב שואל אותך איזה ערך אתה רוצה שיהיה למשתנה L.
(INPUT L).

☐ הקש את המספר (גודל הריבוע) ואל תשכח ENTER.

המחשב שרטט שטיח בדיוק במידה שבחרת!

כיצד קרה הדבר?

כאשר המחשב הגיע לשורה: 12 INPUT L הוא נעצר, הציג על המסך סימן-שאלה, וחיכה שתכניס לו את הערך של המשתנה L.
לאחר שהכנסת מספר, ולחצת ENTER הוא שומר בזכרונו כי המשתנה L שווה למספר זה והוא ממשיך בביצוע התוכנית.
בכל מקום בתוכנית בו מופיע המשתנה L, הוא שווה למספר שהכנסת ב-INPUT.

משימה

שנה את התוכנית כך, שבסיום של הדפסת כל שטיח, המחשב ישאל אותך, באופן אוטומטי, פעם נוספת מה הגודל בו אתה רוצה את השטיח הבא. כל זאת - בלי למחוק את השטיח הקודם שכבר צויר. כמובן, שלאחר שהכנסת את ערכו של L, דאג לכך שהשטיח הקודם יימחק ובמקומו יודפס שטיח חדש בגודל הרצוי. (ושוב מופיע סימן-שאלה של ה-INPUT על המסך).

(תשובה 26 בעמוד 65)

שכלול מדהים!

רוצה שהמחשב גם ישאל אותך מאיזה סימן אתה רוצה שהוא יבנה את השטיח? (במקום $)

☐ הכנס את השינויים הבאים בתוכנית:

- רשום שורת INPUT נוספת:

```
14 INPUT A$
```

- והחלף בשורה 40 את ה-"$" ל-A$.
(ללא מרכאות).

☐ לאחר הכנסת התוספות הרץ את התוכנית.
סימן שאלה מופיע על המסך: המחשב שואל אותך לגודל השטיח.
בחר בגודל ולחץ ENTER.

עכשיו מופיע עוד סימן-שאלה - המחשב רוצה לדעת מאיזה סימן לבנות את הריבוע. בחר סימן כלשהו ולחץ שוב ENTER - הממושמע משרטט שטיח בגודל ובצורה שפקדת עליו!

לפני שנכנס להסבר מפורט - שחק להנאתך עם התוכנית:

- הכנס גדלי-שטיח שונים.
- הכנס בכל פעם סימן אחר.
- הכנס גם סימנים גרפיים..

נהנית? נעבור להסבר יותר מפורט:

משתנה מחרוזת

בשורה 12 יש משתנה L שקלט את המספר שהכנסת:

```
12 INPUT L
```

בשורה 14 (ובשורה 4Ø), לעומת זאת, מופיע משתנה מסוג חדש הנקרא משתנה מחרוזת:

```
14 INPUT A$
```

מהי מחרוזת?

למען האמת, כבר השתמשת בעבר במחרוזות, מבלי לדעת שהן "מחרוזות":

מה מבצע המחשב בפקודה הבאה: PRINT "2+5"

המחשב אינו מבצע את פעולת החשבון אלא פשוט מעתיק את כל מה שכתוב בתוך המרכאות.

זכור: כל דבר שתשים בתוך מרכאות יחשב כמחרוזת:

"מחרוזת"

בפקודה הבאה:

```
PRINT "4*7=" ; 4*7
```

איזה חלק הוא מחרוזת ואיזה חלק הוא ביטוי חשבוני?

(תשובה 27 בעמוד 65)

מכמה סימנים מורכבת כל אחת מהמחרוזות הבאות:

"IBM"

"*25/7^"

(תשובה 28 בעמוד 65)

ומה יקרה אם תרצה שהמחשב ידפיס את המחרוזת הבאה

"BASIC"

ובטעות פקדת עליו:

PRINT BASIC

כמובן שהמחשב הבין זאת כשם של משתנה מספרי ורשם את הערך Ø.

ומהו משתנה מחרוזת?

אם הבנת מהו משתנה-מספרי, לא יקשה עליך להבין מהו משתנה-מחרוזת:

כתוב במחשב:

LET B$ = "XØR" (אל תמחק את התוכנית שישנה בזכרון)

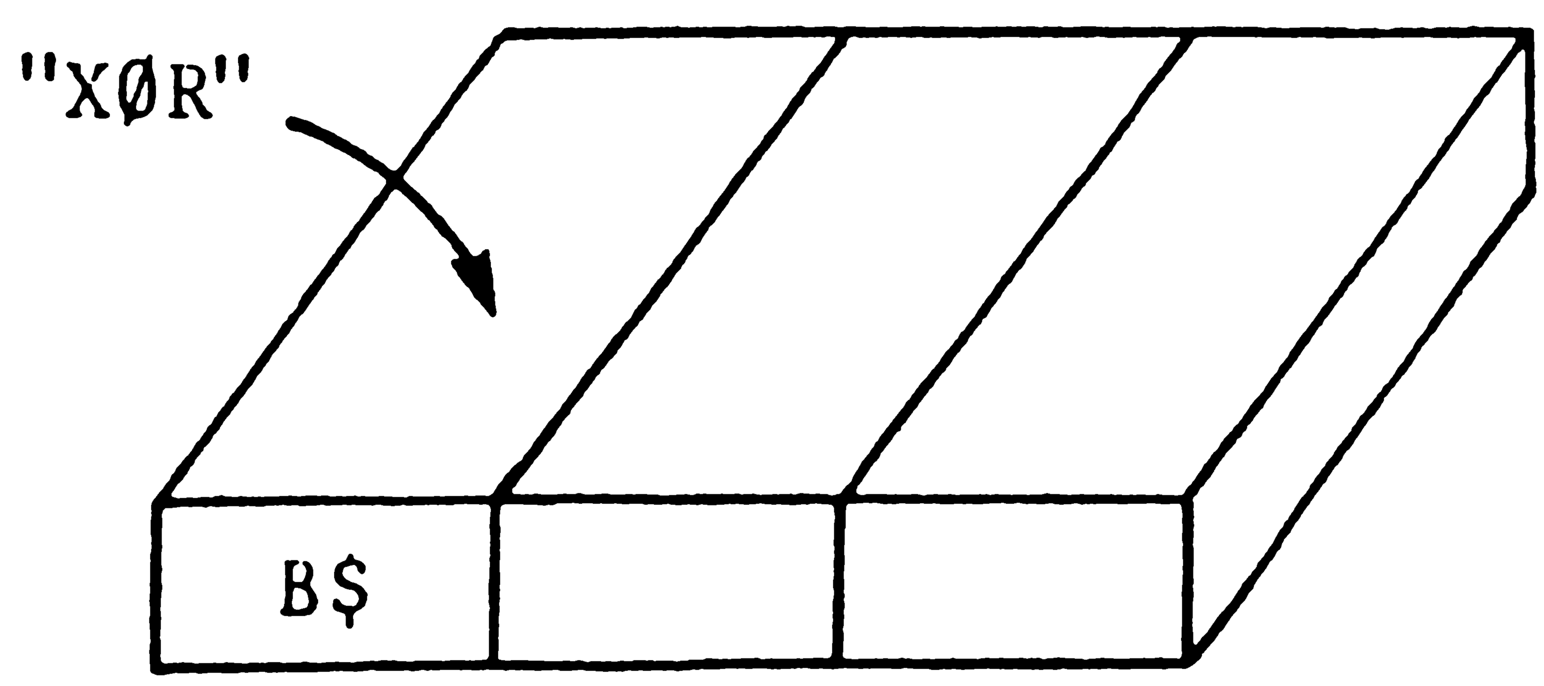
מה יקרה אם תפקוד:

PRINT B$

נסה. כיצד תסביר את התוצאה?

התהליך דומה מאוד למה שקורה במחשב עם משתנים מספריים:

המחשב קרא לאחד התאים שבזכרוננו: B$ והכניס לתוכו את המחרוזת "XØR"



כאשר פקדת PRINT B$ המחשב מצא את התא הנקרא B$, העתיק מתוכו את המחרוזת והדפיס אותה על המסך.

מה יקרה, לדעתך, אם תכתוב: LET B$ = 5

נסה ותיווכח אם צדקת.

המחשב הודיע לך TYPE MISMATCH - כלומר: אין התאמה בין השם של התא בזכרון (B$ - משתנה מחרוזתי) לבין מה שניסית להכניס לתוכו (5 - ערך מספרי).

האם המחשב יקבל LET B$ = "5"? נסה!

נסה להכניס מחרוזת לתא של משתנה מספרי (למשל LET B="5").

מה תעשה תוכנית שכזאת:

```
500 X$ = "שמר"
510 FOR N=1 TO 10
520 PRINT X$
530 NEXT
```

כדי לבדוק אם אתה צודק אל תמחק את התוכנית הנמצאת כבר בזכרון!

העתק את התוכנית, כתוב RUN 500 ולחץ ENTER (אל תשכח רווח אחרי RUN). המחשב יתחיל "לרוץ" משורה 500 ואילך, וכך תוכל לבדוק אם צדקת לגבי התוכנית הקטנה הכתובה למעלה.

מחיקת מספר שורות מהתוכנית - DELETE

כדי לעשות זאת פקוד על המחשב:

DELETE 500-530 (יש קיצור ל-DELETE)

לאחר שלחצת על ENTER המחשב מוחק באופן אוטומטי מזכרונו את כל השורות שבין 500 לבין 530.

הצג את התוכנית ותיווכח.

שמות של משתני מחרוזת

הכללים החלים על שמות של משתנים מספריים נכונים גם לגבי משתני-מחרוזת:

- השם חייב להתחיל באות אנגלית.
- אחרי אות זו יכולות לבוא אותיות וספרות כרצונך.

אבל:

שם של משתנה מחרוזת חייב להסתיים תמיד בסימן $!

מתוך השמות הבאים של משתני מחרוזת - איזה חוקיים ואיזה המחשב לא יקבל:

A$, 2A$, AAA$, B3, C$$, C*$

(תשובה 29 בעמוד 65)

המחרוזת הריקה

זוכר מה קרה כאשר ביקשת מהמחשב להדפיס משתנה מספרי שלא הגדרת קודם לכן את ערכו?

הוא פשוט הדפיס Ø.

ואיך יגיב המחשב על משתנה מחרוזת שלא הוגדר?

נחש מה יופיע על המסך כאשר תפקוד על המחשב:

```
PRINT "A" ; ZX$ ; "B"
```

ד	ג	ב	א
A☐B ↑ רווח	AZX$B	AB	AØB

לאחר שניסית לנחש...

כתוב "B" ; XZ$; "A" PRINT (ולחץ ENTER).

הסבר

ZX$ הוא משתנה-מחרוזתי שלא הוגדר עדיין (לא קיבל כל ערך קודם). במקרה כזה - כאשר אתה לא נותן למשתנה מחרוזתי ערך כלשהו - הוא מקבל את הערך של המחרוזת הריקה:

ZX$ = ""

מחרוזת ריקה היא מחרוזת שאינה מכילה כלום בין המרכאות - אפילו לא רווח (ובודאי לא Ø).

וחזרה לתוכנית שלנו:

בתוכנית שלנו רצינו שהמחשב ישרטט את הריבוע עם אחד מהסימנים - אות, מספר או צורה גרפית.

לשם כך השתמשנו במשתנה מחרוזת A$.

כאשר המחשב הגיע לשורה:

```
14 INPUT A$
```

הוא מציג סימן-שאלה על המסך ומחכה שתגיד לו למה אתה רוצה שמשתנה המחרוזת A$ יהיה שווה.

שים לב: אינך צריך, במקרה זה, לרשום מרכאות - המחשב יודע כי עליו לקלוט מחרוזת. (איך הוא יודע?)

לאחד שהקשת על אחד הסימנים (עם ENTER, כמובן), המחשב יודע כי משתנה המחרוזת A$ שווה לסימן שהקשת.
בכל פעם שהוא מגיע לשורה

```
40... : PRINT A$
```

הוא מדפיס את הסימן שתקתקת ל-INPUT A$.

מה יקרה אם משתנה המחרוזת A$ יהיה שווה למחרוזת בת 2 סימנים? (למשל: A$ = "AB")

נסה!

כותרות ל־INPUT

לתוכנית שלך יש עכשיו שתי שורות עם פקודות INPUT:

- עבור המשתנה המספרי L.
- עבור המשתנה המחרוזתי A$.

שמת בודאי לב, כי סימני-השאלה שהמחשב מציג הם זהים.
כדי לא להתבלבל בין שני ה-INPUT אפשר להורות למחשב להדפיס כותרת מיוחדת לכל סימן-שאלה שהוא מציג, וכך נדע בדיוק לאיזה נתון הוא מחכה.

הנה דוגמא:

שנה את שורה 12:

```
12  INPUT "L=" ; L
```

המשתנה נשאר L ← L

הכותרת היא מחרוזת — "L="

אל תשכח — ;

- הרץ את התוכנית.

עכשיו מתקבל: L = ?

ואתה יודע בדיוק לאיזה משתנה המחשב מחכה.

- עצור את ריצת התוכנית.

עכשיו הכנס כותרת גם ל-INPUT השני.

לדוגמא:

```
14 INPUT "SIMAN=" ; A$
```

הערה:

ודאי שמת לב כי -

```
PRINT "2+5=" 2+5
```

זהה לחלוטין ל-

```
PRINT "2+5=" ; 2+5
              ↑
```

כלומר אפשר להשמיט את הנקודה-פסיק

אבל -

אי אפשר להשמיט את הנקודה-פסיק מן הכותרת של ה-INPUT.

- נסה למשל -

```
14 INPUT "SIMAN=" A$
```

- והרץ.

RENUM

- הצג את התוכנית שפיתחת על המסך.

אתה רואה כי בגלל ששתלת מספר שורות חדשות בין שורה 1Ø ל-2Ø, הן "צפופות", ויתכן שבהמשך פשוט לא יהיה מקום לשתול שורות נוספות.

המחשב מאפשר לך למספר באופן אוטומטי את כל השורות מחדש, בעזרת הפקודה RENUM:

- פקוד על המחשב לבצע את הפקודה RENUM.
- לאחר שתלחץ ENTER הצג פעם נוספת את התוכנית.

המחשב מיספר את השורות מחדש בקפיצות של 10 אחת מהשניה.
שים לב לשורה בה מופיעה פקודת ה-GO TO.

לאיזו שורה מופנה עכשיו המחשב ________ (12, 2Ø).

מסקנה

כאשר המחשב מבצע את פקודת ה-RENUM, הוא משנה אוטומטית את מספרי השורות המופיעים בפקודות ה-GO TO.
במקרה שלנו הוא שינה את שורה 12 ל-2Ø, לכן גם GO TO 12 הפך ל: GO TO 2Ø.

הוסף שורות לתוכנית כך שהמחשב ידפיס את שתי הוראות ה-INPUT בחלקו העליון של המסך, ולא מתחת ל"שטיח"...

(תשובה 30 בעמוד 65)

סיימת?
פקוד על המחשב למספר מחדש את התוכנית.

לסיום...

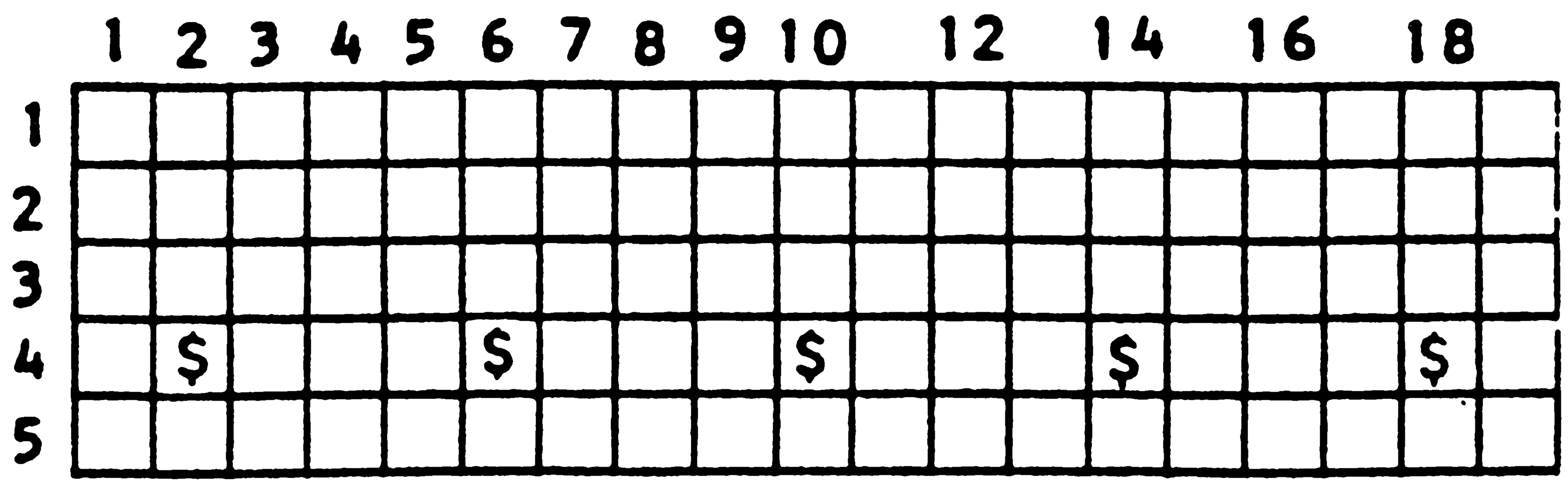
הצג את התוכנית שפיתחת על המסך.
אתה רואה כי כבר בשלב זה התוכנית שפיתחת היא די מורכבת.
תוך שימוש במושג המשתנה ובמושג הלולאה למדת להשתמש בשני מושגים חשובים חדשים:

- ה-INPUT
- משתנה המחרוזת.

פרק ה
שכפול רובוטים

שמוש ב־STEP

יש לכתוב תוכנית אשר תשכפל 5 דולרים במסך Ø:



☐ השלם את התוכנית הבאה כדי שהיא תעשה זאת:

```
5   WIDTH 4Ø : CLS
1Ø  X = ___
2Ø  FOR N=1 TO ___
3Ø  LOCATE ___,___ : ________
35  FOR T=1 TO 2ØØ : NEXT T
4Ø  X = ___ + ___
5Ø  NEXT N
```

☐ הכנס את התוכנית למחשב והרץ אותה.

(תשובה 31 בעמוד 65)

קיצור התוכנית

ניתן לקצר באופן משמעותי את התוכנית בעזרת ההוראה **STEP** (**סְטֶפּ**) שפירושה - **צעד**.

שנה את התוכנית הנמצאת בזכרון באופן, שתתקבל התוכנית:

```
5   WIDTH 4Ø : CLS
2Ø FOR X=2 TO 18 STEP 4
3Ø LOCATE 4, X : PRINT "$"
35 FOR T=1 TO 2ØØ : NEXT T
4Ø NEXT X
```

שים לב: גם אחרי STEP צריך רווח!

הרץ את התוכנית. האם קיבלת אותה תוצאה כמו בתוכנית הקודמת?

מה יקרה לדעתך אם תשנה את STEP 4 ל-STEP 2?

הכנס את השינוי וראה אם צדקת.

נסכם:

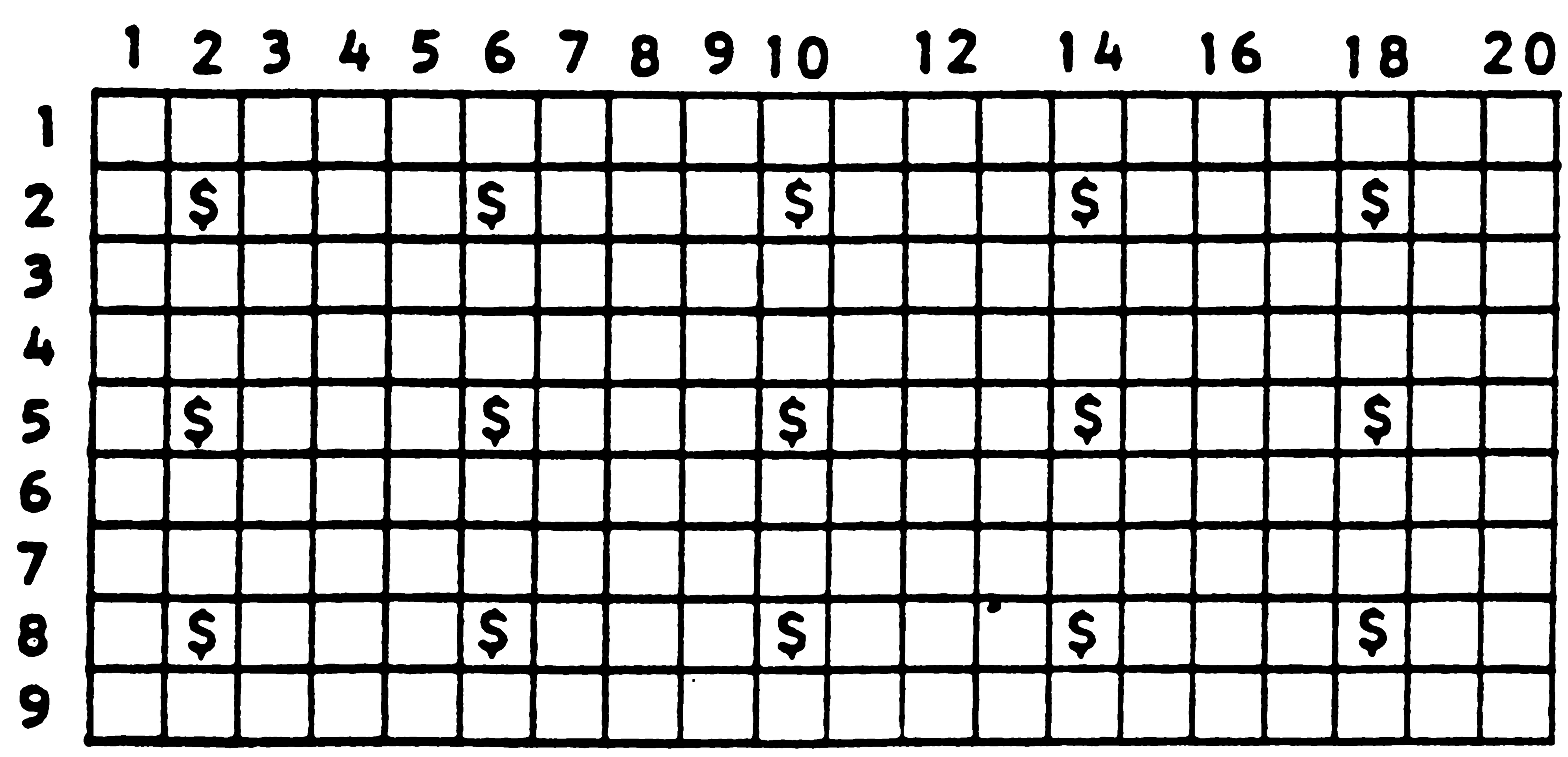
הוספת STEP נותנת הוראה למחשב לשנות את המשתנה המופיע ב-FOR-NEXT "בקפיצות" (צעדים) בהתאם למספר הרשום אחרי STEP:

STEP 4 אומר למחשב כי על משתנה הלולאה לגדול כל פעם ב____.

STEP 2 אומר למחשב כי על משתנה הלולאה לגדול כל פעם ב____.

משימה

הוסף עוד שתי שורות לתוכנית כך שהיא תדפיס 3 שורות של $ ברווחים של שתי שורות ריקות בין שורה לשורה. כך:



(תשובה 32 בעמוד 66)

מחק את התוכנית ועבור הלאה...

מה תעשה התוכנית הבאה:

```
10 FOR N=1 TO 40 STEP 5
20 PRINT N
30 NEXT N
```

לאחר שחשבת - הרץ את התוכנית ובדוק עצמך.

ומה יקרה לדעתך בתוכנית הבאה:

```
10 FOR N=100 TO 0 STEP -10
20 PRINT N
30 NEXT N
```

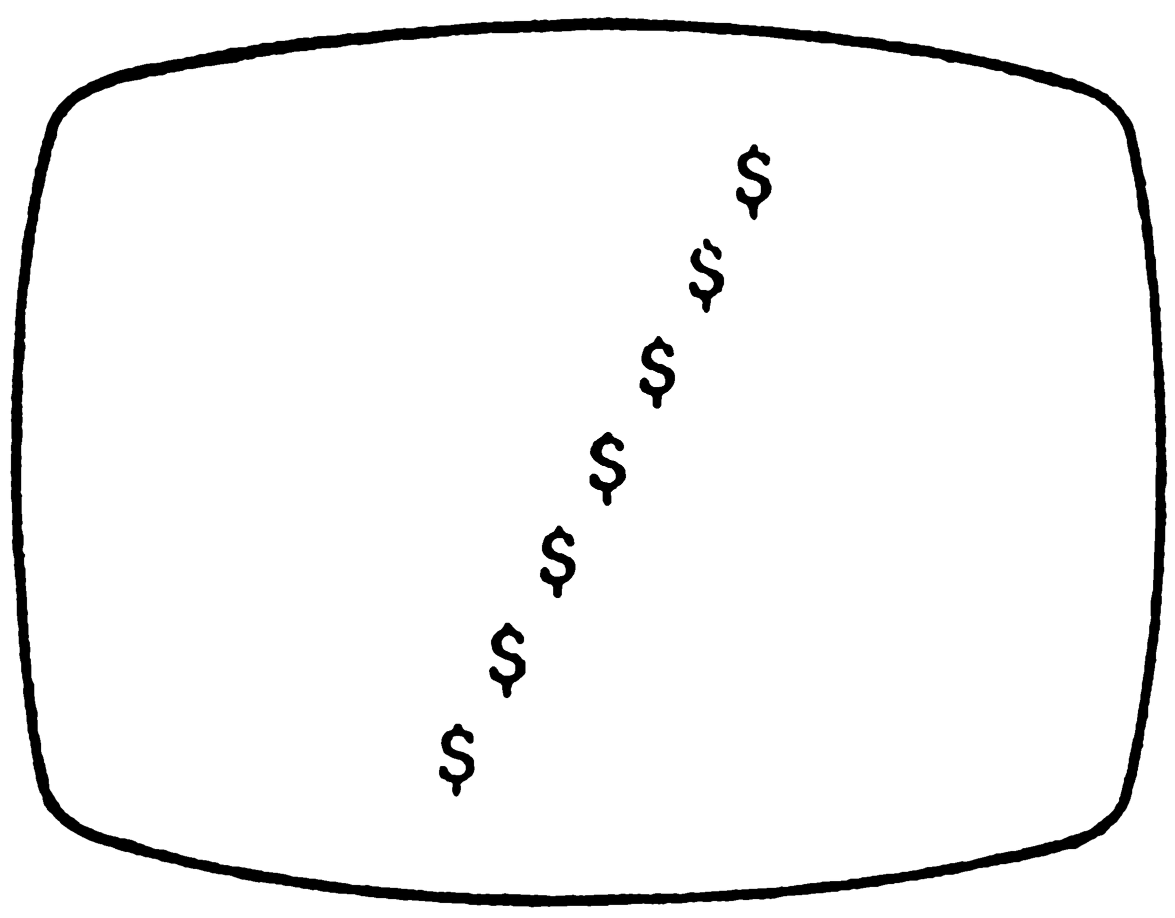
(שים לב לסימן המינוס)

הרץ אותה במחשב.

הפעם הלך המחשב אחורנית: הוא התחיל ב-100 והגיע עד 0.

משימה

השתמש ב-STEP שלילי כדי לשרטט אלכסון של דולרים במסך 0:



(תשובה 33 בעמוד 66)

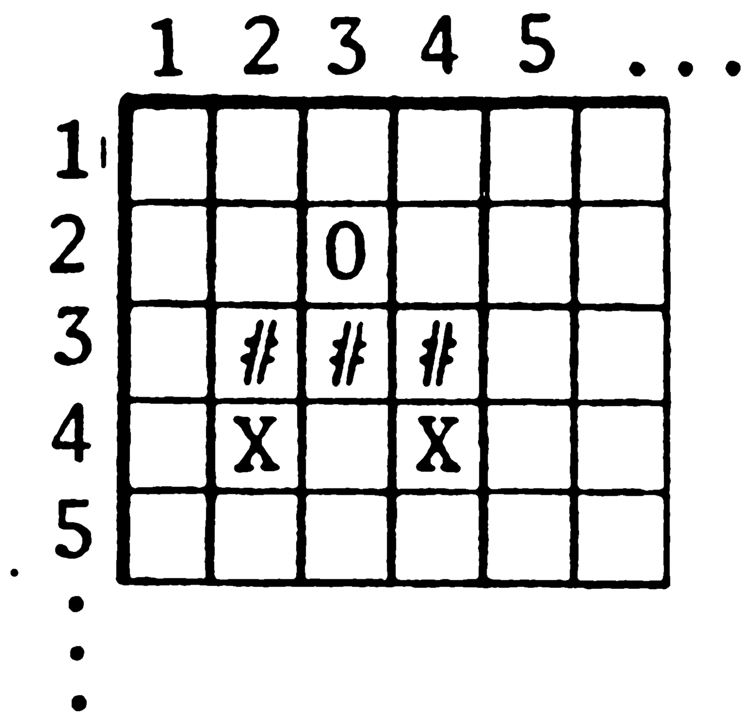
שנה את התוכנית כך, שכל $ יתקבל בצבע אחר...(אם ברשותך מחשב צבעוני)

(תשובה 34 בעמוד 66)

משימה כתוב תוכנית שתשמיע קול של סירנה:
צליל עולה ויורד שוב ושוב.

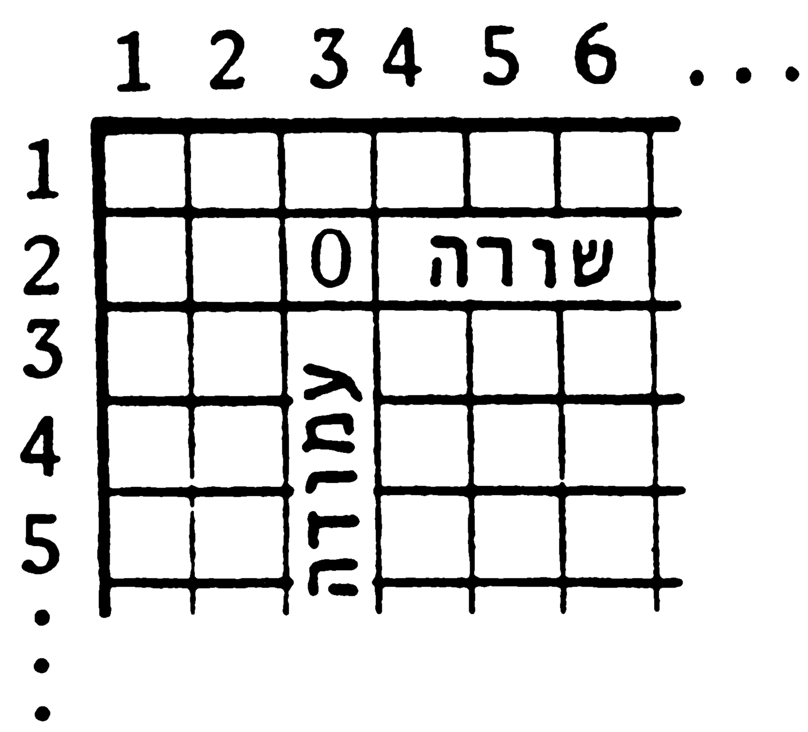
שכפול רובוטים

נא הכר את אי-לי אחיו הצעיר
של הרובוט אי-ציק מהיחידה
הקודמת:
(מיד יופיעו על המסך גם
אחיו התאומים.)



נתחיל בכתיבת התוכנית המציירת את אי-לי:

א. השלם את השורות הבאות על-מנת לקבל את ראשו של אי-לי
בדיוק במשבצת הנמצאת בעמודה 3 ובשורה 2:
הערה: הראש בנוי מן
האות "O"




```
5   WIDTH 4Ø : CLS
1Ø  Y = __________
2Ø  X = __________
3Ø  LOCATE Y, X : __________
```

אנו כותבים את אי-לי באופן כללי, בעזרת המשתנים Y
ו-X, כדי שנוכל בהמשך לשכפל אותו.

ב. עכשיו צריך להוסיף את הכתפיים והגוף:
שים לב כי הכתפיים מתחילות בשורה אחת ___ (מעל, מתחת)
לראש, ועמודה אחת ____ (שמאלה, ימינה) מהראש.
לכן, הפקודה לקבלת הכתפיים והגוף תהיה:

```
4Ø LOCATE Y+ ___, X- ___ : PRINT ____
                  ↑
```

(שים לב! זה סימן מינוס)

השלם את השורה והכנס אותה למחשב. בדוק אם מתקבלת

התוצאה הרצויה.

אם אתה מתקשה פנה לתשובה 35 בעמוד 66.

אגב: האם Y ו-X משנים את ערכם לאחר שהמחשב מבצע את שורה 4Ø?

ג. נעבור לרגליים:

הוסף שורה שתצייר את הרגליים:

5Ø ____________________

הרץ ובדוק אם אתה מקבל את אי-לי על המסך.

(תשובה 36 בעמוד 66)

משימה

שנה את התוכנית כך שהמחשב ישאל אותך באיזה שורה (Y) ובאיזה עמודה (X) אתה רוצה שיופיע ראשו של אי-לי. ומיד הוא ישרטט, באופן אוטומטי, רובוט נוסף בשורה Y ובעמודה X - מבלי למחוק את הרובוט הקודם שכבר הופיע.

מיד לאחר שהמחשב משלים את שרטוט הרובוט הנוסף הוא חוזר ושואל אותך למקום חדש של רובוט נוסף...

נסה להתגבר על המשימה בכוחות עצמך.

אם אתה מתקשה, נאמר לך, כי אחד המושגים שלמדת עליהם בפרק הקודם יבואו לעזרתך.

סיימת לכתוב ולהריץ את התוכנית? -

השווה לתוכנית שבתשובה 37 בעמוד 66.

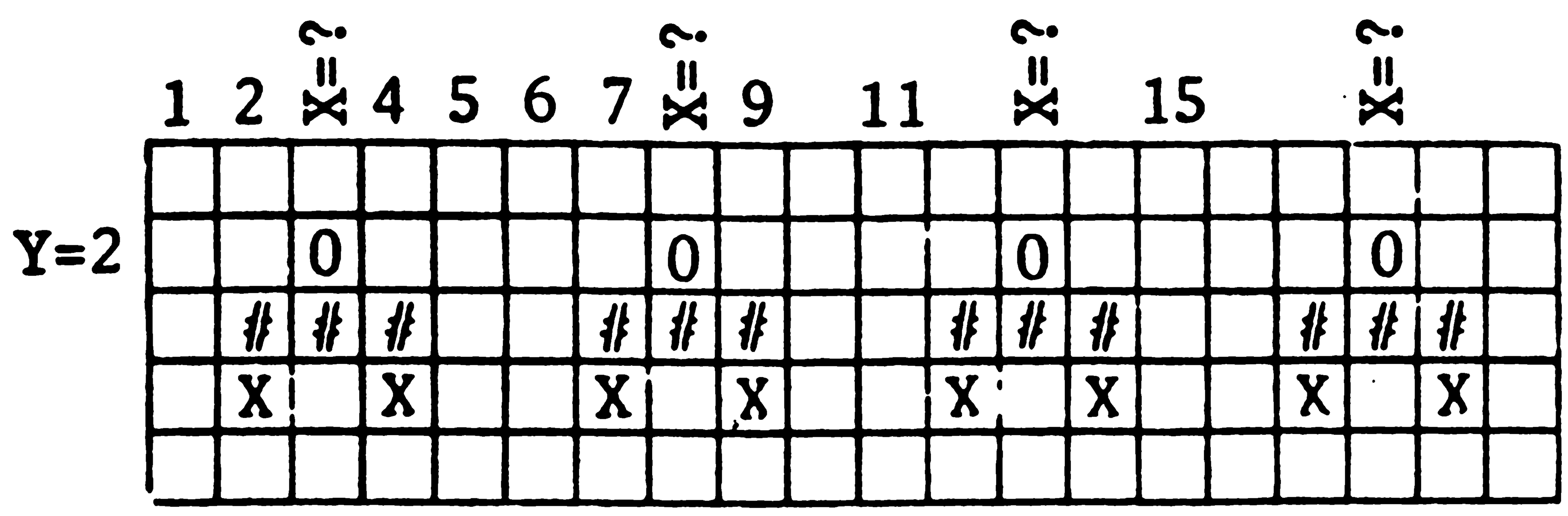
שכפול אוטומטי של אי־לי:

עד כה גרמת אתה למחשב לשכפל את אי-לי, לאחר שהכנסת ב-INPUT ערכים למשתנה Y ולמשתנה X.

האם תוכל לפתח תוכנית שתשכפל באופן אוטומטי מספר שורות של אי-לי?

נתחיל בשורה אחת:

עליך להוסיף ולשנות את התוכנית כך שהמחשב ידפיס את אי-לי 4 פעמים לרוחב המסך במקומות הבאים:



(רמז: עליך להשתמש ב-FOR-NEXT וב-STEP)

(תשובה 38 בעמוד 66)

ועכשיו הוסף פקודות כך שהתוכנית תשכפל שלוש שורות של אי-לי זו מתחת לזו.

(חשוב - כיצד קיבלנו את השורות בעמוד 41?).

(תשובה 39 בעמוד 66)

ועוד אתגרים...

- שכפל את אי-לי באלכסון.
- שכפל אותו בשני אלכסונים.

CHR$

עכשיו שנה את השורה האחראית להדפסת הראש ל:

```
30 LOCATE Y, X : PRINT CHR$ (2)
```

הרץ את התוכנית.

ובכן, אם עבדת נכון המחשב הדפיס את הרובוט עם פרצוף מחייך!

כיצד קרה הדבר?

לכל סימן, אות, מספר וצורה גרפית יש מספר סידורי מסוים השמור באופן קבוע בזכרונו של המחשב.

פתח את ספר ה-BASIC של היצרן בנספח G ‏(APPENDIX G). בנספח זה מופיעה רשימה מפורטת של מספריהם הסידוריים של כל הסימנים וכן שרטוט של כל סימן וסימן.

זהה את הצורה הגרפית שמספרה 2.

מה יקרה לדעתך לרובוט, אם במקום לבחור בסימן שמספרו 2 תבחר בסימן 1?

נסה!

האם הרובוט החליף פרצוף?

ומה יקרה אם במקום 1 תבחר בסימן שמספרו הסידורי הוא 79?

נסה!

נסכם:

CHR$ (79) היא מחרוזת המכילה את האות "O".

CHR$(79) = "O"

מבחינתו של המחשב "O" ו-CHR$(79) הם פשוט אותו הדבר.

החזר לרובוט את פרצופו הצוחק.

שנה את גופו של הרובוט

עכשיו שנה את גופו של הרובוט כך שבמקום הסימן # הוא יהיה מורכב מסימן גרפי שמספרו 177.

שים לב!

מאחר ש-CHR$(177) היא מחרוזת לכל דבר תוכל לגרום למחשב להדפיס שלושה סימנים כאלה בזה אחר זה בעזרת נקודה פסיק (;).

(תשובה 40 בעמוד 66)

שנה את רגליו

עליך לבחור שני סימנים גרפיים מתאימים שיחליפו את האות X של הרובוט.

(עיין בסימנים השונים ובחר בצורה או בצורות המתאימות.)

(תשובה 41 בעמוד 66)

צבע את הרובוט

הוסף שורות מתאימות לתוכנית כך שכל רובוט יופיע בצבע שונה.

(אל תשכח לשנות את מספורן של השורות בעזרת RENUM אם יש צורך בכך.)

משימה

כתוב תוכנית, שבה המחשב ישאל אותך מה מספרו של הסימן אותו אתה מעונין לראות - ויראה לך אותו.

(תשובה 42 בעמוד 67)

בדוק גם את הסימנים הבאים: 171, 153, 11, 7

האם שמת לב, כי 7 הוא מספרו הסדורי של סימן קולי?

המחשב משמיע צפצוף קצר במקום להציג סימן כלשהו על המסך.

משימה

כתוב תוכנית שתציג את כל הסימנים, החל מהסימן שמספרו הסידורי הוא 33.

CHR$(13)

חלק ממקשי הפקודות (F) כוללים בתוכם את המקש ENTER.
עכשיו נלמד כיצד להוסיף את המקש ENTER לתוך מקשי-הפקודות.
אם תתבונן ברשימת הסימנים ב-APPENDIX G תראה כי המספר 13 מייצג את המקש ENTER (הנקרא גם CARRIAGE RETURN).
נניח כי ברצוננו לשנות את F1Ø כך, שבלחיצה עליו ינקה המחשב את המסך.

☐ פקוד: `KEY 1Ø, "CLS" + CHR$(13)`

☐ בדוק אם אכן המחשב מנקה את המסך מבלי שתצטרך ללחוץ על מקש ENTER.
אתה רואה כי הוספת למעשה את ההוראה ENTER לפקודה CLS.

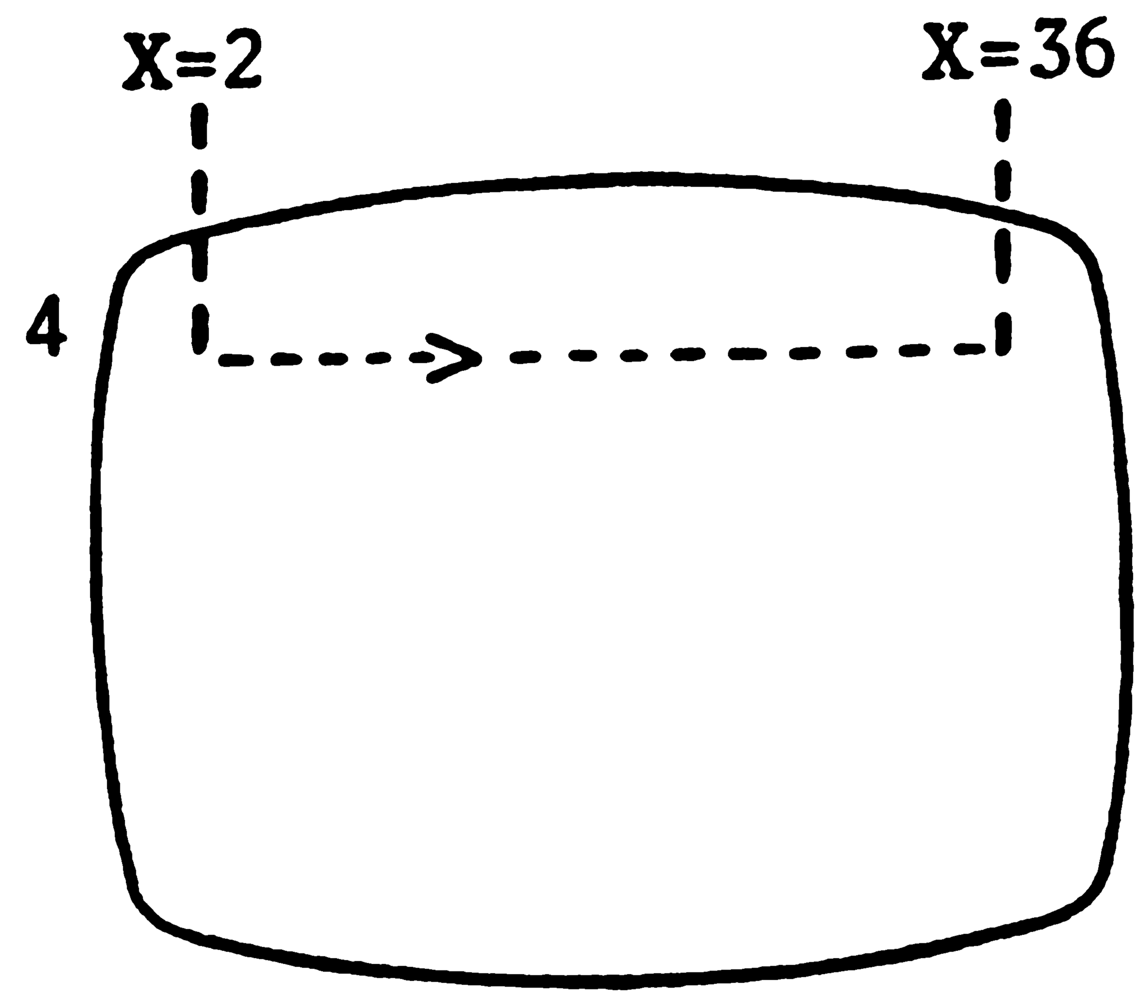
☐ התבונן במלבן העשירי בתחתית המסך (אם אתה ברוחב 8Ø). קרוב לודאי שעכשיו מופיע שם: ← CLS. כבר אמרנו כי החץ ← מצביע על כך ש-ENTER בוסף לפקודה.

פרק ו
אנימציה – יצירת תנועה על המסך

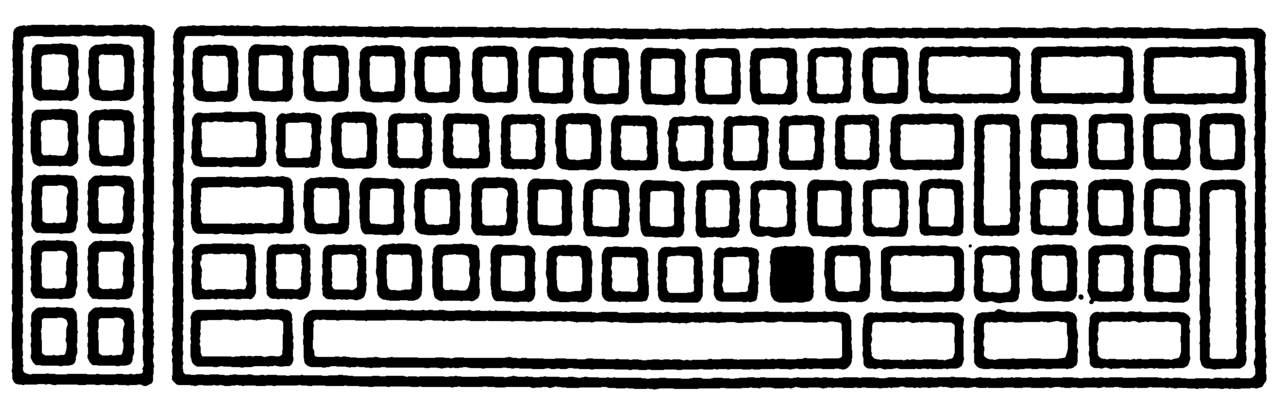
זוכר את אי-ציק עושה התעמלות בוקר ביחידה הראשונה?
עכשיו ננצל את הכלים החדשים שרכשת בינתיים כדי להניע
צורות על המסך.

משימה 1

עליך לכתוב תוכנית אשר תגרום לצורה < "לעוף" לרוחב המסך
(עבוד ברוחב 4Ø), מעמודה 2
עד עמודה 36 בשורה 4:
כאשר הצורה < תגיע לעמודה 36
היא תתחיל שוב לנוע משמאל
לימין וכן הלאה.



אגב, את הצורה < תמצא על
לוח המקשים כאן:

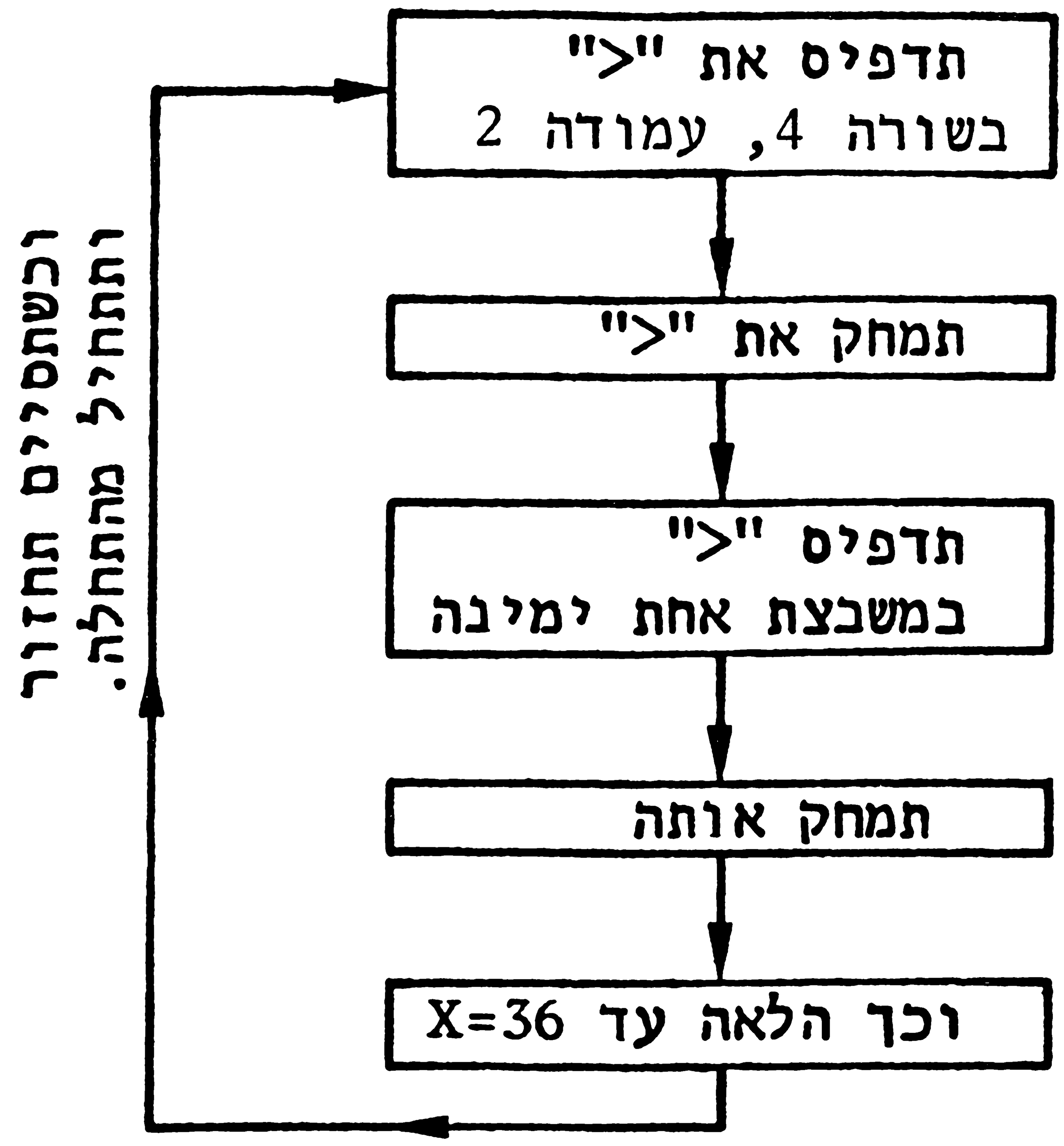


דע לך כי אתה יודע את כל הפקודות הדרושות.
רוצה לנסות לבד? - נסה עכשיו לכתוב את התוכנית ולהריצה.

אם אתה מתקשה כדאי שתפנה לרמז בעמוד הבא:

רמז

הנה מבנה התוכנית: עליך לכתוב תוכנית אשר



האם מתוך מבנה התוכנית תוכל כבר לכתוב את התוכנית בעצמך? -

- נסה!
- הרץ את התוכנית. אם מצאת "בגים" בתוכנית - היה סבלני ונסה לתקן אותם.

הערה "הציפור" טסה מהר מדי?

האט אותה בעזרת לולאת השהייה.

(תשובה 43 בעמוד 67)

משימה 2

עליך לשנות את התוכנית כך שתחילה תראה הצורה - "<" בעה משמאל לימין (יוצאת "מקיר" 1 ומגיעה ל"קיר" 2)

■ > ■

"קיר" 2 "קיר" 1

כאשר הצורה מגיעה ל"קיר" 2 היא תתהפך: ">" ותנוע בחזרה מ"קיר" 2 ל"קיר" 1

■ < ■

2 1

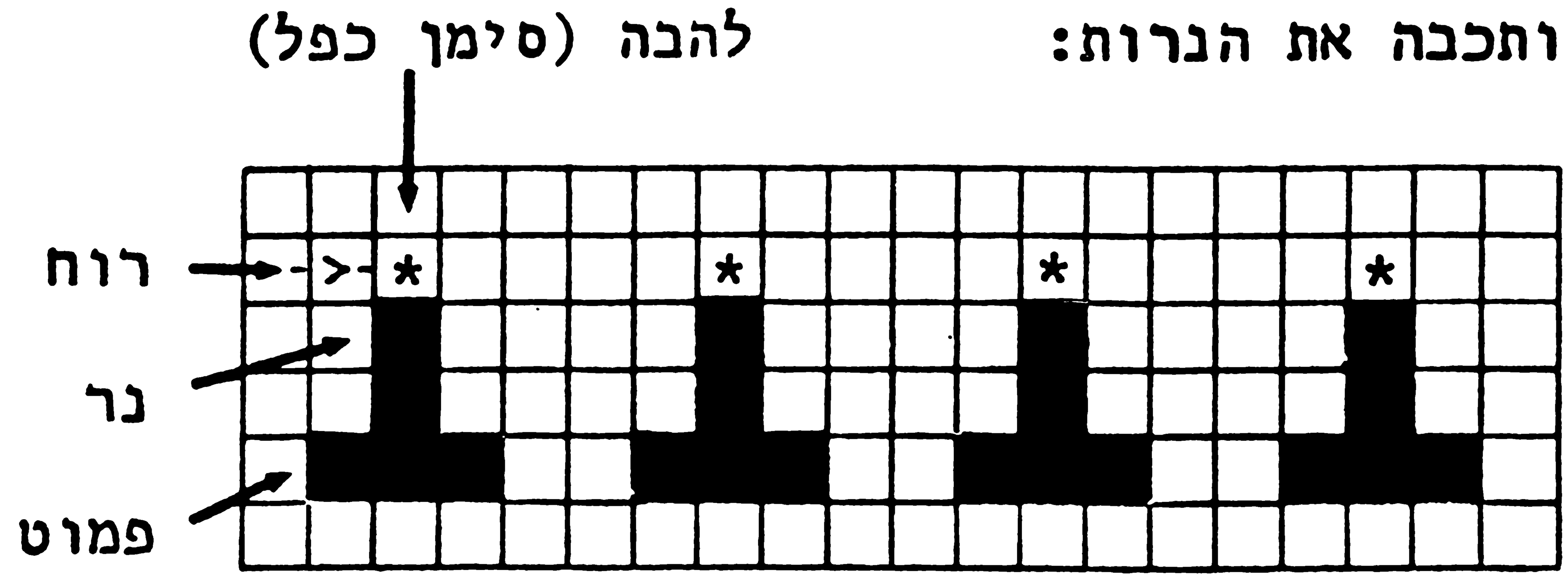
- את "הקירות" בנה מהסימן "■" (מספרו הסידורי הוא 219).
- את הצורה ההפוכה "<" תמצא על לוח המקשים.

בבנית התוכנית הקפד:

- "הציפור" אינה מוחקת את "הקירות".
- "הציפור" עפה הלוך ושוב ללא הפסק.
- הקירות אינם מהבהבים.
- הציפור מגיעה במעופה ממש עד הקירות.
- צבע את הקירות, את הציפור ואת הרקע כרצונך.
(אולי תהיה מעונין לשנות את צבע הציפור והקירות בכל כיוון?) (תשובה 44 בעמוד 67)

משימה 3

עליך לכתוב תוכנית שתצייר 4 נרות (כמו באיור).
לאחר שהנרות דולקים, תופיע הרוח הרעה (<), תנוע ימינה ותכבה את הנרות:



הנה כמה עצות:

- הרוח תנוע כמו הציפור בתוכנית הקודמת.
- הלהבות בנויות מסימן הכפל * .
- כתוב את התוכנית המשכפלת את הנרות, כמו שעשינו עם אי-לי בפרק הקודם.
הכנס צבעים כרצונך!

לאחר שהתמודדת לבד, תוכל לפנות ולהשוות ל:

(תשובה 45 בעמוד 67)

רוצה לקבל "נרות-פלא"?

הוסף פקודה שתיצור לך נרות פלא: בכל פעם שהרוח גומרת לכבות אותם - הם נדלקים באופן אוטומטי, והרוח מנסה לכבות אותם שוב ושוב...

פרק ז
מסגרות לפי מידה

בפרק זה עליך לפתח תוכנית אשר תבצע את הפעולות הבאות:

- המחשב שואל אותך איזה שם אתה רוצה להכניס למסגרת.
- אתה כותב את השם - ואז המחשב כותב אותו במרכז המסך (בערך) ומצייר מסביבו מסגרת המתאימה את עצמה באופן אוטומטי לגודל השם שבחרת!

לדוגמא:

```
* * * * * * * *
*             *
*   D A N I   *
*             *
* * * * * * * *
```

ועבור שם ארוך יותר - מסגרת ארוכה:

```
* * * * * * * * * * *
*                   *
*   K A L I S K Y   *
*                   *
* * * * * * * * * * *
```

LEN - אורך המחרוזת

כדי לכתוב את התוכנית עליך להכיר את ההוראה LEN:

LEN הוא קיצור של המילה LENGTH שפירושו - **אורך**.

כדי להבין את פעולת ההוראה LEN:

פקוד על המחשב

```
PRINT LEN ("ABCDE")
```

(אל תשכח את הסוגריים)

לאחר שלחצת ENTER המחשב כתב את המספר: ______.

ומספר זה שווה בדיוק למספר הסימנים המופיעים במחרוזת - זהו אורך המחרוזת.

? מה יכתוב המחשב על המסך לאחר ביצוע התוכנית הבאה:

```
20 X1$ = "A1B2C3"
30 PRINT LEN (X1$)
```

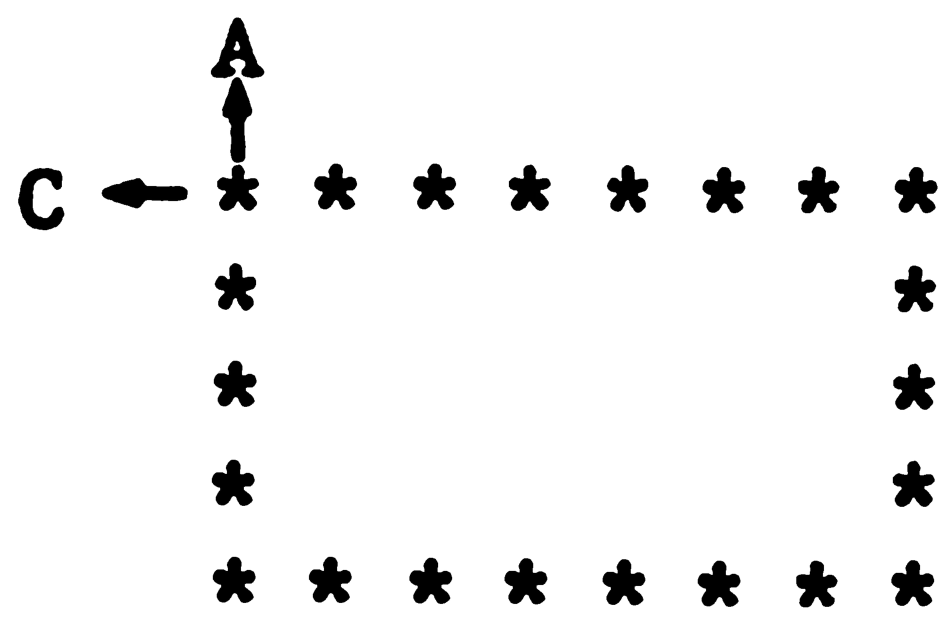
נסכם:

אתה רואה כי פקודת LEN אומרת למחשב לספור את מספר הסימנים שבתוך המחרוזת.

ועכשיו - לאחר שהכרת את פקודת ה-LEN, אתה יכול להשתמש בה על-מנת לבנות את תוכנית המסגרות. אנו מציעים כי תעיז ותנסה להתמודד עם התוכנית לבדך - אפילו אם הדבר יגזול ממך זמן מה. אם אתה בכל זאת מתקשה - נפתח את התוכנית שלב אחר שלב:

שלב 1

- ראשית כל עליך לדעת לכתוב תוכנית המציירת מסגרת מלבנית, באורך מוגדר וברוחב מוגדר ללא קשר לשם שיופיע בתוכו:



אורך המלבן המופיע כאן הוא 8 כוכביות ורוחבו - 5 כוכביות.

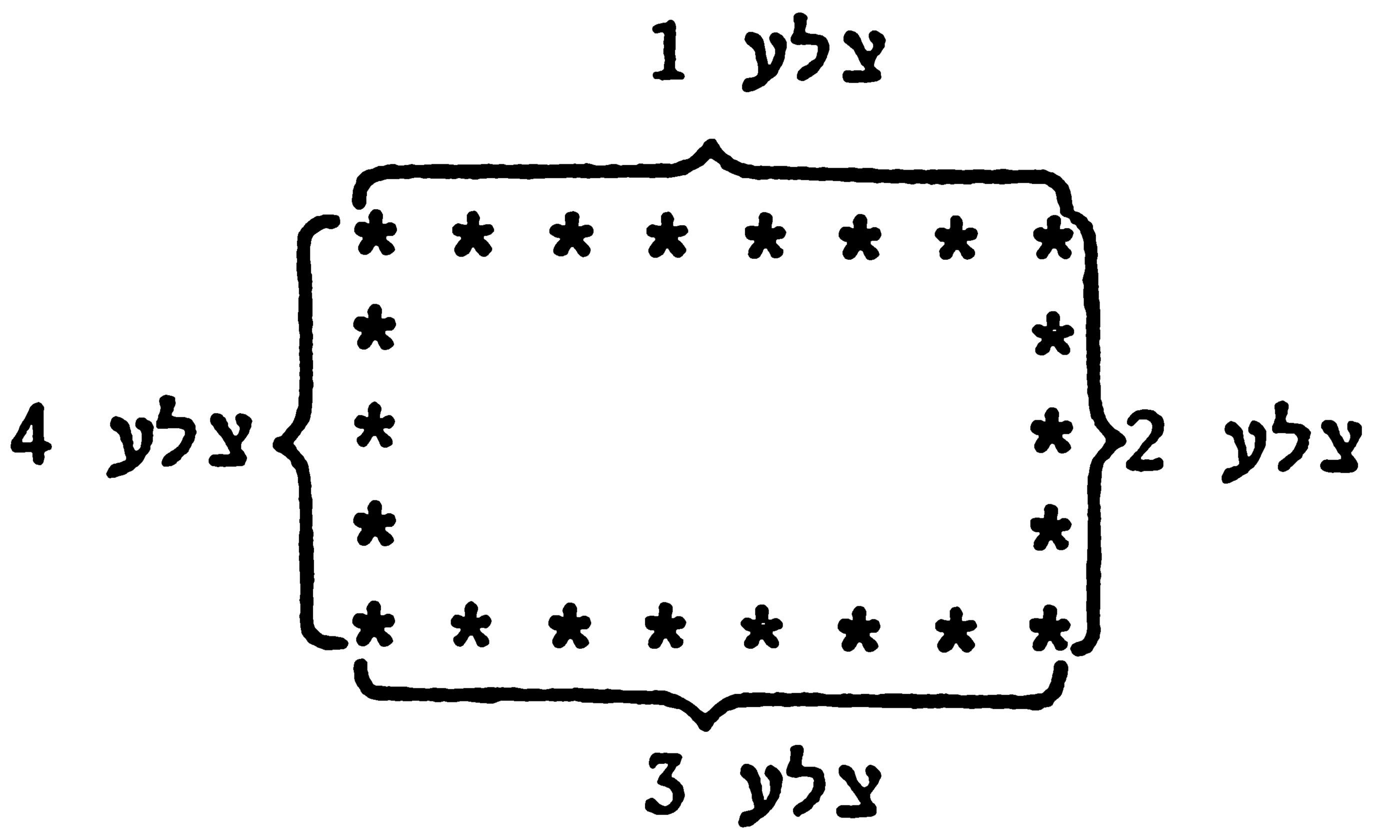
- עליך לכתוב תוכנית שתשרטט את המלבן כאשר הכוכבית השמאלית העליונה במצאת בשורה 3 (C=3) ובעמודה 3 (A=3).

כיצד תעשה זאת?

ישנם מספר דרכים:

יתכן כי תרצה לשרטט תחילה את צלע 1, לאחר מכן 2, 3 ולבסוף 4, או בתחילה את 1 ו-3 ולאחר מכן 2 ו-4.

בחר בדרך הנוחה לך.



(תשובה 46 בעמוד 68)

מה עליך לעשות בתוכנית כך שהמחשב ישאל אותך באיזה <u>אורך</u> מלבן אתה מעונין והמחשב ישרטט אותו באופן אוטומטי.

(תשובה 47 בעמוד 68)

שלב 2

כעת נעבור לכתיבת השם עצמו.

- עיקר הקושי בתוכנית הוא להתאים את אורך המסגרת לאורך השם:

הנה דוגמא:

```
* * * * * * * *
*             *
*   D A N I   *
*             *
* * * * * * * *
```

אורך המסגרת

השם מורכב מ-4 סימנים.

המסגרת מורכבת מ-___ (4, 8) סימנים.

לכן: אורך המסגרת = אורך השם + ___ (4, 8, 2)

תוכל להשוות את תוכניתך לזו בתשובה 48 בעמוד 68.

האם הודעת OK מפריעה לך?

- הוסף שורה כך שהודעה זאת לא תופיע על המסך לאחר כתיבת השם במסגרת.
- הוסף גם שורה כזאת לתוכנית, שהמסגרת תשאר לבדה על המסך, ללא הודעת ה-INPUT.

(תשובה 49 בעמוד 68)

שכלולים

- המחשב ישאל אותך גם מאיזה סימן לבנות את המסגרת.

- המחשב ישאל אותך על איזה צבע רקע אתה מעונין שכל מסגרת תופיע.

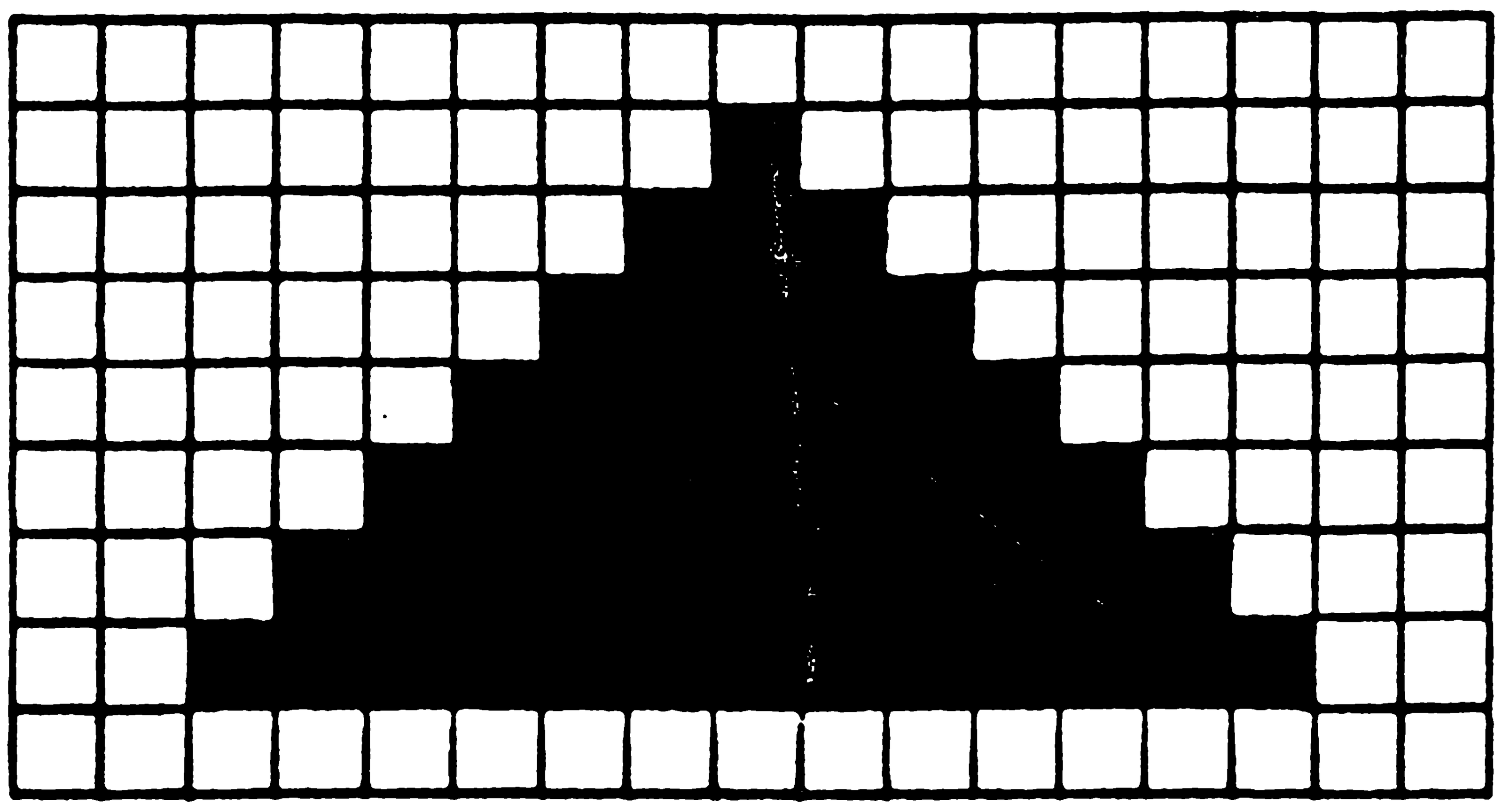
- המחשב ישאל אותך באיזה צבע לכתוב.

פרק ח
בנית צורות

עצה ללומד: אם תמצא פרק זה כקשה ל"עיכול", אל תנסה "לבלוע" אותו בשלב זה. קפוץ לעמוד 61 לסיכום ומשם ליחידה 3. בהמשך תוכל לחזור לפרק זה ולהתמודד איתו בשנית וביתר הצלחה.

משימה 1: בנית "פירמידה"

עליך לכתוב תוכנית, שתשרטט את הפירמידה הבאה:



התוכנית צריכה להיות בנויה כך, שהיא תכלול רק פקודת LOCATE אחת.

אתה מתקשה? בעזרת התרגילים הבאים תוכל לפענח את סוד בנית הפירמידה:

תרגיל 1

מה עושה התוכנית הבאה:

```
1Ø WIDTH 8Ø : CLS
2Ø A=15 : B=25 : Y=5
3Ø FOR X=A TO B
4Ø LOCATE Y, X : PRINT CHR$(219)
5Ø NEXT X
```

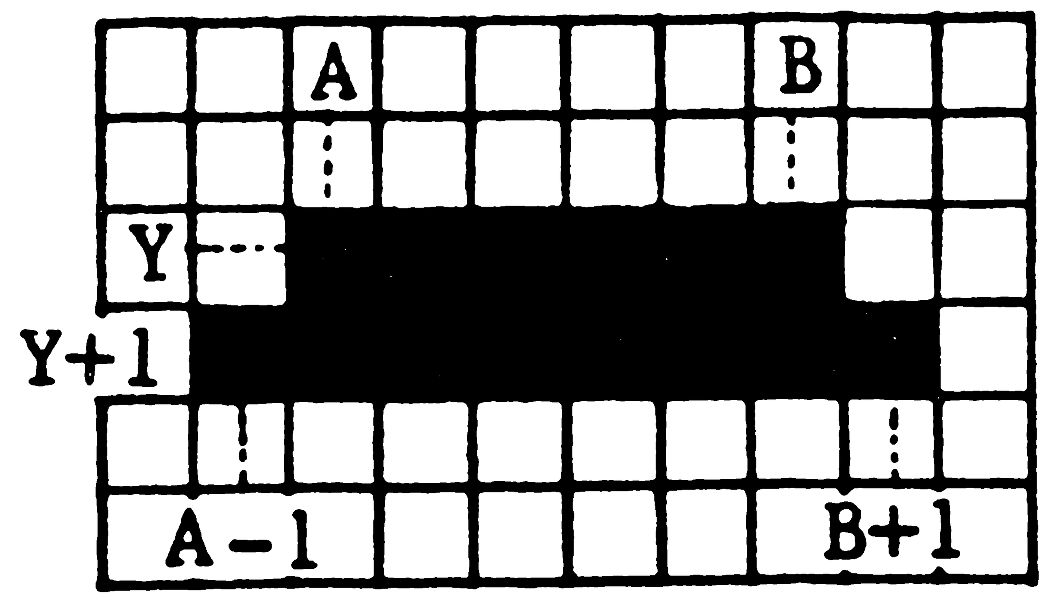
אחרי שחשבת - בדוק עצמך במחשב. נסה כעת לכתוב את תוכנית הפירמידה.

תרגיל 2

אם עדיין לא הצלחת - הנה תרגיל עזר נוסף:

עליך לשנות ולהוסיף פקודות לתוכנית שבתרגיל 1 כך,

שהתוכנית תרשום גם את השורה שבתרגיל 1, ותוסיף עוד שורה:



ועכשיו, לאחר שבצעת את שני תרגילי העזר בהצלחה, אתה חייב לדעת כיצד לכתוב את התוכנית של הפירמידה.

ואם בכל זאת לא הצלחת
ראה תשובה 50 בעמוד 68

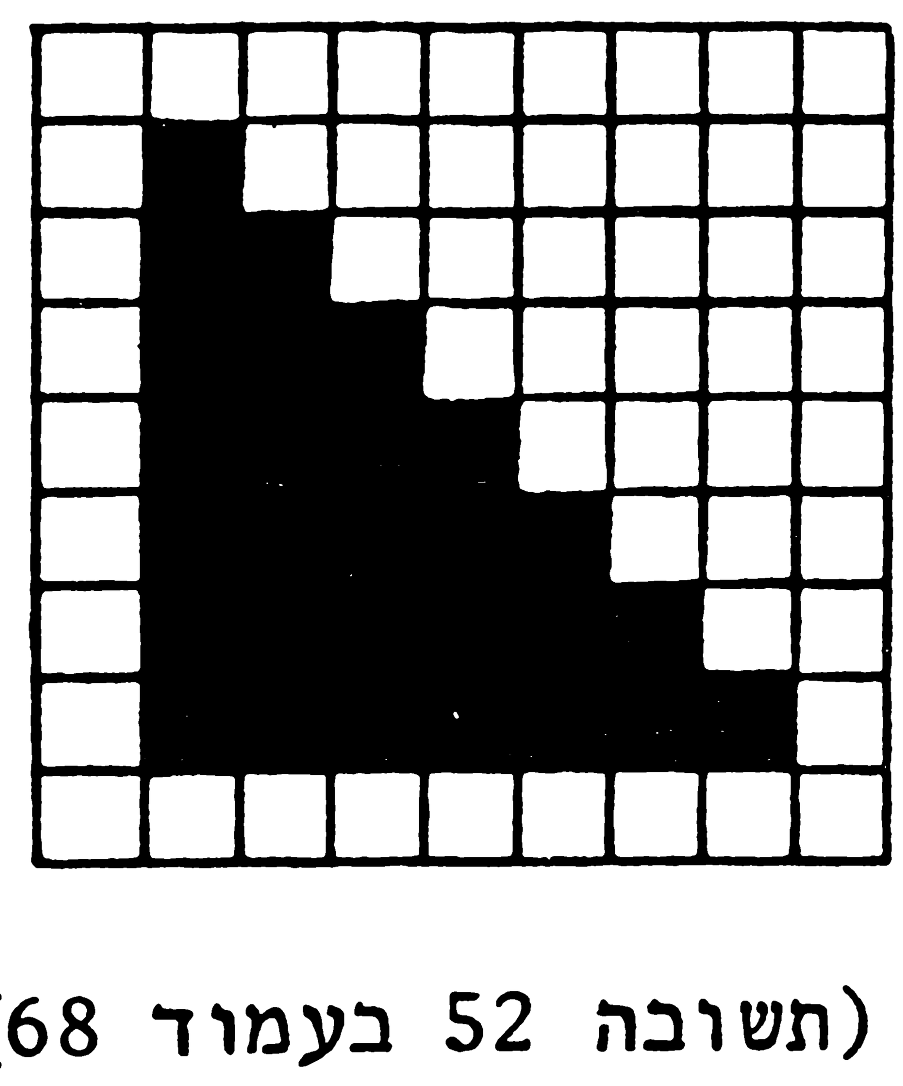
משכללים את התוכנית

הוסף לתוכנית שינוי קטן כך, שלפני ציור הפירמידה המחשב ישאל אותך מאיזה סימן לבנות אותה.

(תשובה 51 בעמוד 68)

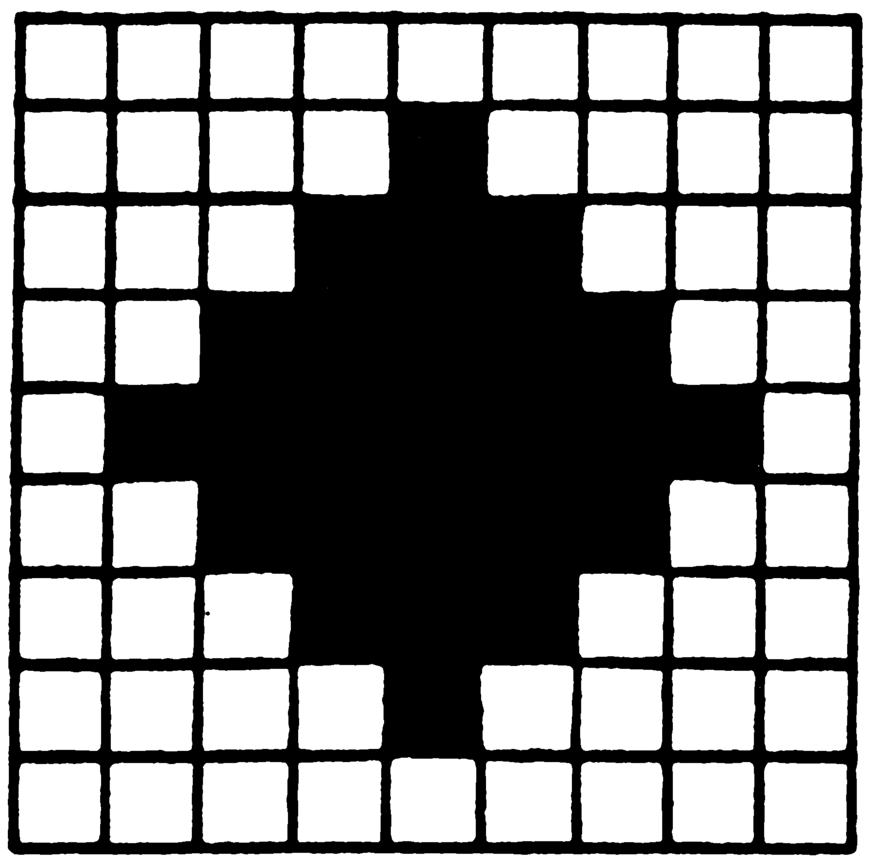
הוסף צבעים וצלילים כרצונך.

מה עליך לעשות בתוכנית כדי לקבל צורה כזאת?



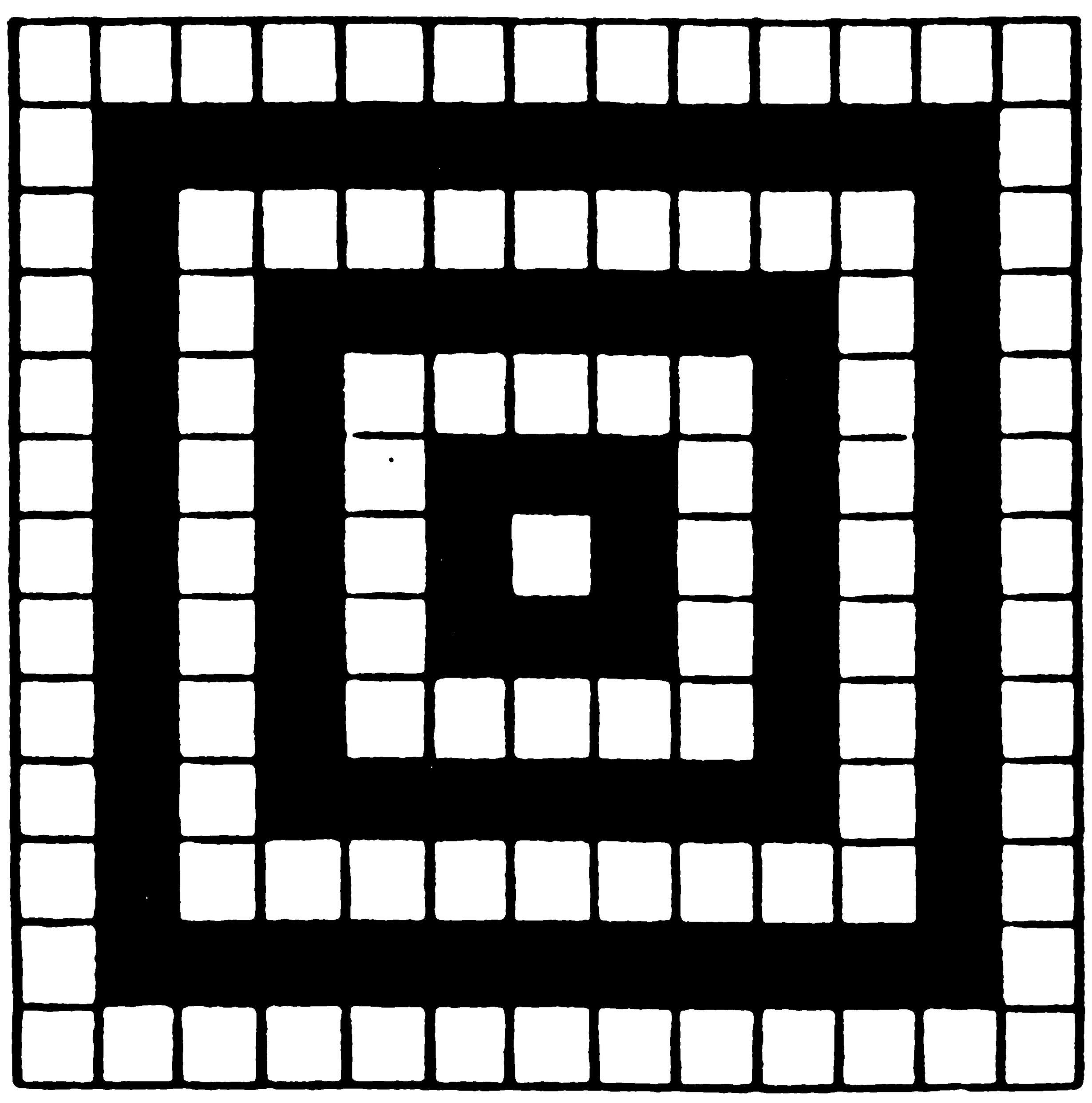
(תשובה 52 בעמוד 68)

האם אתה מעונין לבנות צורה כזאת?
אנו משאירים אותך להתמודד עם
זה לבדך...



משימה 2: "ריבועים מתנפחים"

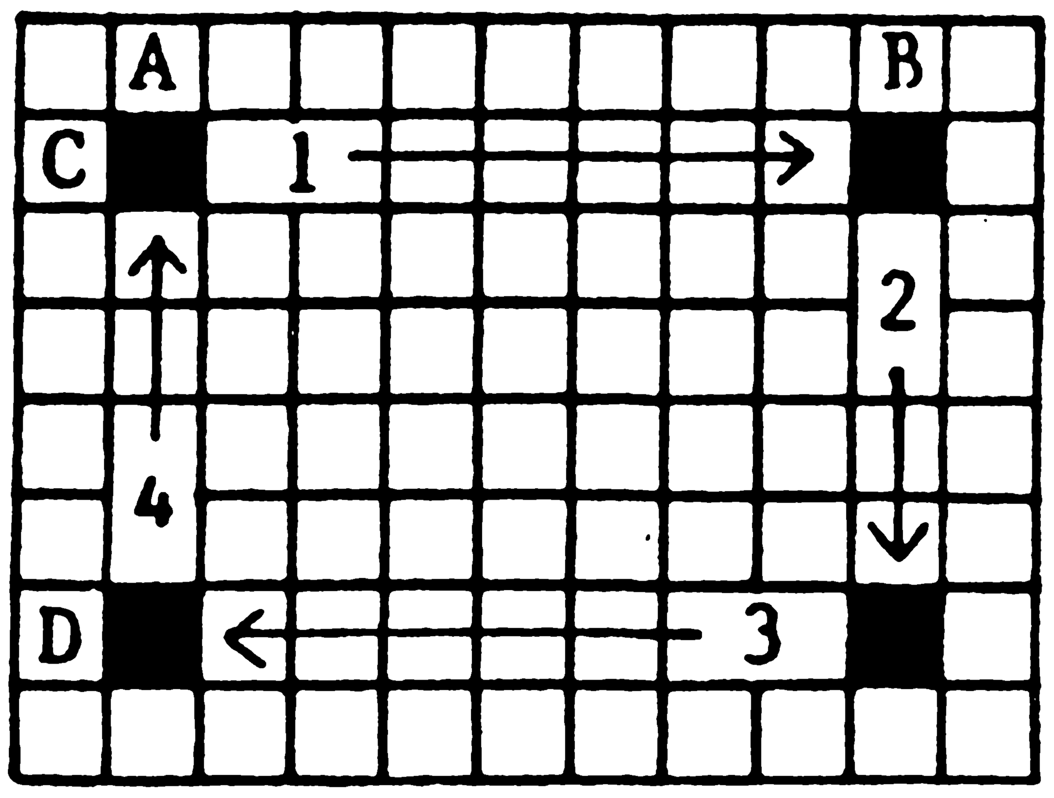
עליך יהיה לכתוב תוכנית, אשר תצייר מלבנים זה בתוך זה:
בהתחלה תצייר התוכנית את המלבן הפנימי, לאחר מכן את שאר
המלבנים בזה אחר זה:



ושוב, נסה בכוחות עצמך לכתוב את התוכנית.

עצה מועילה: על-מנת לבנות את התוכנית, עליך לחשוב בתחילה
כיצד לבנות מלבן אחד כללי. אח"כ עליך להוסיף פקודות
שישכפלו את המלבן וינפחו אותו.
אם אתה מתקשה, פנה לתרגילי העזר שבהמשך.

תרגיל עזר



עכשיו נבנה תוכנית לבניית
מלבן כללי בין העמודות A
ו-B לבין השורות C ו-D.

נבנה את התוכנית בשלבים:

שלב א

כתוב תוכנית, בה ישרטט המחשב את צלע 1 משמאל לימין.
תן ערכים ל-A, B ו-C כרצונך.
הרץ את התוכנית במחשב. (תשובה 53 בעמוד 69)

שלב ב

הוסף לתוכנית פקודות, שבעזרתן ימשיך המחשב וישרטט את צלע 2
מלמעלה למטה. (תן ערך גם ל-D)
הרץ את התוכנית. (תשובה 54 בעמוד 69)

שלב ג

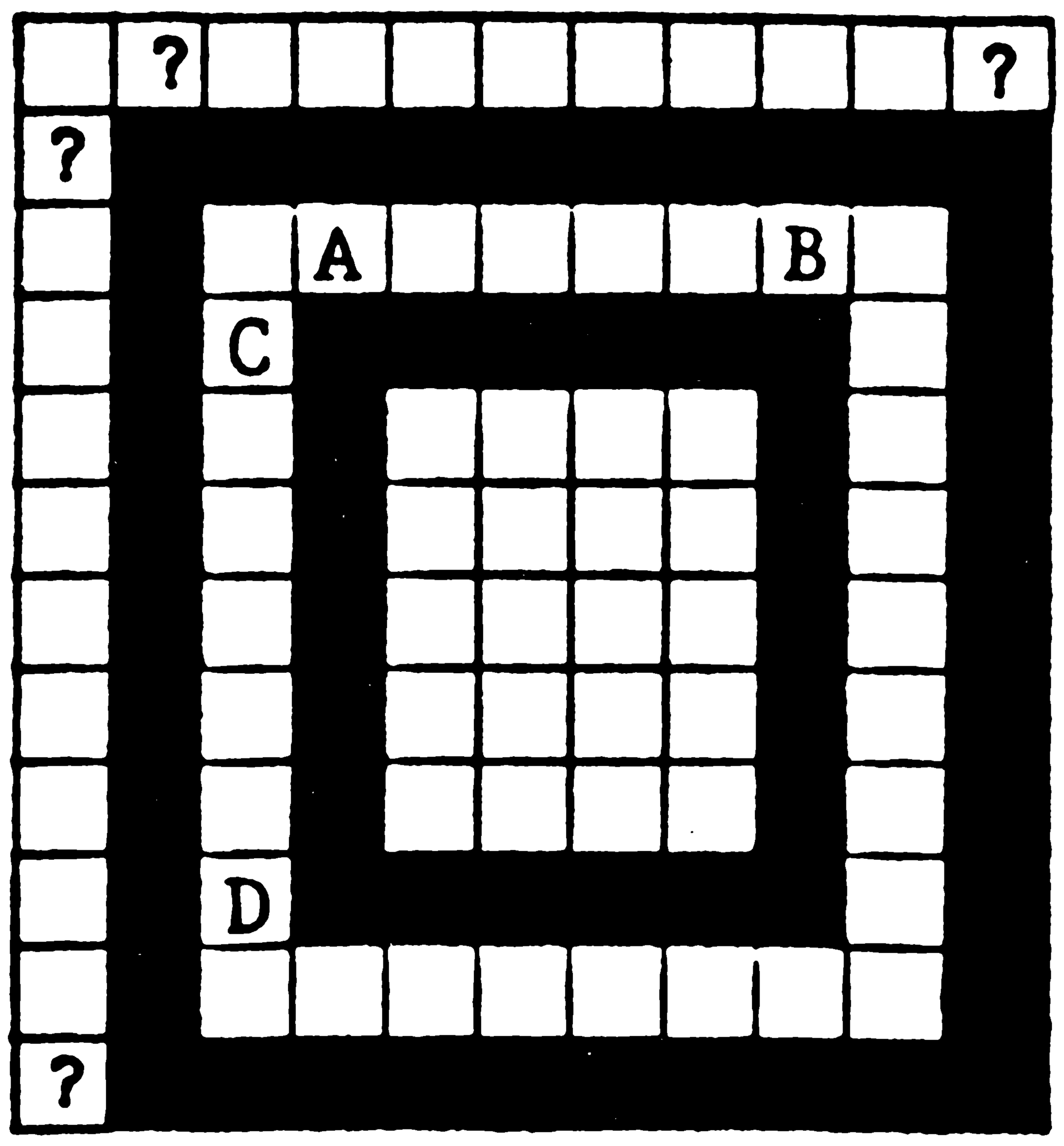
עכשיו הוסף לתוכנית את הפקודות המתאימות כך, שהיא תשרטט את
צלע 3 מימין לשמאל.
הערה: עליך להשתמש ב-1- STEP (שלילי).
הרץ את התוכנית על המחשב. (תשובה 55 בעמוד 69)

שלב ד

ועכשיו לא יקשה עליך להשלים ולכתוב את הצלע הרביעית.
כתוב אותה במחשב והרץ את התוכנית.
הפעם לא ניתן לך תשובה. אם לא הצלחת לשרטט את הצלע
הרביעית, סימן שלא הבנת משהו קודם. על כן, חזור והתחל
את תרגיל העזר מהתחלה.

שלב ה

מה צריך לקרות ל-A, B, C ו-D
כדי שהתוכנית תצייר 3 מלבנים
נוספים?
הכנס את הפקודות המתאימות כך,
שתקבל את שלושת המלבנים
הנוספים.



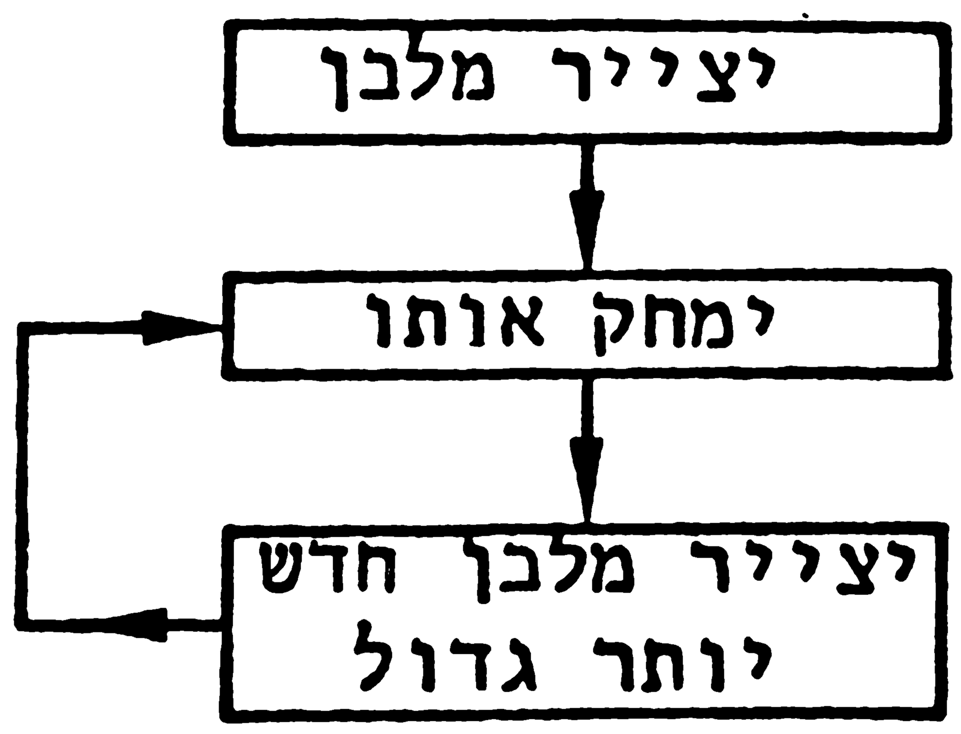
(תשובה 56 בעמוד 69)

הרץ את התוכנית גם ברוחב 8Ø.

בוא "נשחק" עם התוכנית

1. הוסף צבעים למלבנים ולרקע.

2. עכשיו, לאחר שהתוכנית הכללית הסתיימה, הוסף לה פקודה כך, שהמחשב ישרטט את המלבנים באופן הבא:

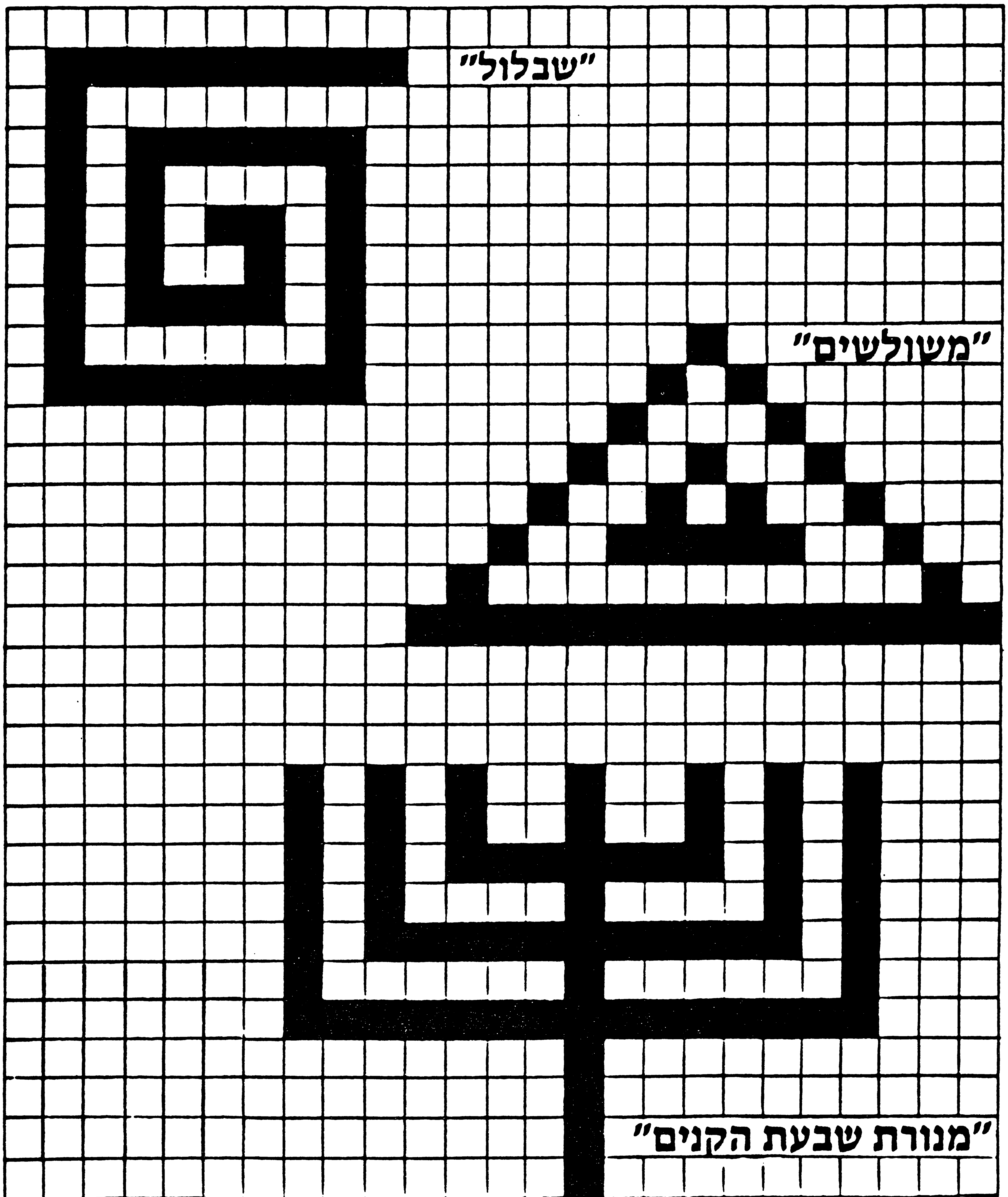


3. • שנה את גודל המלבנים ומיקומם.
   • שנה רק את B והשאר את A, C ו-D קבועים.
   הרץ את התוכנית וראה מה קיבלת.
   וכך המשך לשחק עם שאר המשתנים.
4. המחשב בונה את המלבנים מסימן שאתה בוחר.

לפניך מספר הצעות לשרטוט צורות במחשב

התכניות הן בהישג ידך. יחד עם זאת דרושה סבלנות רבה לכתוב אותן. אם אתה מעונין נסה אותן:

(אל תשכח שלמחשב שלך יש צבע וצליל....)



שאלת חזרה לסכום

לפניך רשימת מושגים שלמדת ביחידה זו.
עבור על כל אחד מהם ובדוק אם אתה זוכר אותם.
(תוכל לרענן את זכרונך בעזרת מספרי העמודים המופיעים בסוגריים ליד כל מושג.):

- משתנה (9)
- שמות של משתנים (10)
- LET A=A+1 (11)
- KEY (16)
- KEY OFF (17)
- KEY ON (17)
- FOR-NEXT (18)
- משתנה לולאה (20)
- לולאת השהיה (22)
- לולאה אינסופית (24)
- לולאות FOR-NEXT שתולות אחת בתוך השניה (26)
- TIME$ (28)
- INPUT (31)
- מחרוזת (33)
- משתנה מחרוזת (34)
- DELETE (35)
- שמות של משתני מחרוזת (35)
- המחרוזת הריקה (36)
- RENUM (38)
- STEP (41)
- STEP-1Ø (42)
- CHR$ (46)
- CHR$(13) (48)

לסיום...

עברת כברת דרך רצינית. אתה כבר יודע לכתוב תוכנית תוך שימוש נרחב בלולאות FOR-NEXT ומשתנים, וכל זאת תוך ניצול תכונותיו המיוחדות של מחשב ה-IBM-PC.
יתכן שבשלב זה אתה כבר יכול להמריא על כנפי דמיונך.
אם כן, אף אחד לא מונע זאת ממך.
ויתכן שתרצה להמשיך וללמוד את יחידה 3:
"פיתוח משחקי מחשב-טלויזיה", אשר עם סיומה תדע להשתמש במירב הכלים העומדים לרשותך בשפת ה-BASIC.
ביחידה 1 - עמוד 71 תמצא פרטים נוספים על יחידה 3.

תשובות

<u>תשובה 1 מעמוד 9</u>

המחשב מודיע כי ניסית לחלק באפס, כיוון שהמשתנה F עדיין לא קיבל כל ערך ולכן F=Ø.
כמו כן המחשב הדפיס את המספר:

$$1.701412E + 38 = 1.701412*10^{38}$$

זהו המספר הגדול ביותר בו יכול המחשב "לטפל".
שים לב לשיטה המיוחדת אשר בה מציג המחשב חזקות כדי לחסוך מקום ולא לרשום מספר רב של אפסים על המסך.
לדוגמא: 7E+12 שווה ל-$7*10^{12}$ כלומר 7 עם 12 אפסים!

<u>תשובה 2 מעמוד 10</u>

השמות הבאים <u>אינם</u> חוקיים: 2D, 25, PRINT
השם PRINT אמנם מתחיל באות אנגלית אך אינו חוקי כי PRINT היא אחת מפקודות השפה BASIC.

<u>תשובה 3 מעמוד 12</u>

```
1Ø WIDTH 4Ø : CLS
2Ø N=1
3Ø LOCATE 1Ø, 2Ø : PRINT N
35 SOUND 3ØØØØ, 2Ø
4Ø N=N+1
5Ø GO TO 3Ø
```

הערה: ודאי שמת לב כי בהתחלה המספרים מתחלפים במהירות - בהמשך תלמד פסק זמן יעיל יותר!

<u>תשובה 4 מעמוד 13</u>

```
2Ø      N=1            N=1
3Ø... PRINT N          N=1      ┌►N=2       ┌►N=3
4Ø N=N+1               N=2      │ N=3       │ N=4
                       └────────┘ └─────────┘ └────►
5Ø GO TO 3Ø
```

<u>תשובה 5 מעמוד 13</u>

יש לשנות: 4Ø N=N+2

<u>תשובה 6 מעמוד 19</u>

יש לשנות: 1Ø FOR N=1 TO 8

<u>תשובה 7 מעמוד 19</u>

המחשב יבצע את שורות 2Ø, 3Ø ו-4Ø בדיוק 2Ø פעמים.

תשובה 8 מעמוד 19

```
1Ø CLS
2Ø N=1 TO 7
25 B=B+1
3Ø LOCATE B, B : PRINT "A"
4Ø NEXT B
```

תשובה 9 מעמוד 19

```
1Ø CLS
2Ø FOR N=1 TO 5
3Ø LOCATE N, N : PRINT "ISAK"
4Ø NEXT N
```

תשובה 10 מעמוד 21

```
                          RUN
                           ↓
1Ø FOR N=1 TO 3 --- N=1    ┌─→N=2     ┌─→N=3
2Ø PRINT N -------- N=1    │  N=2     │  N=3
3Ø NEXT N --------- N=2────┘  N=3─────┘  N=4
4Ø PRINT "SOF"                            ↓
                                         SOF
```

כש-N מקבל ערך הגדול מ-3 (TO 3) המחשב יוצא מ-FOR-NEXT וממשיך לשורה 4Ø.

תשובה 11 מעמוד 22

```
1Ø CLS
2Ø FOR N=1 TO 7
3Ø LOCATE N, N : PRINT "A"
4Ø NEXT N
```

תשובה 12 מעמוד 22

יש לשנות:

```
2Ø FOR T=1 TO 2ØØØ
```

תשובה 13 מעמוד 23

הסוד נעוץ בשורות

```
2Ø FOR T=1 TO X*X:NEXT T
3Ø X=X+1
```

כיוון ש-X גדל ב-1 בכל פעם, הרי שהמחשב "סופר" בתחילה רק עד 1 (1×1), אח"כ עד 4 (2×2) ואח"כ עד 9 (3×3) וכן הלאה עד מספרים גדולים מאוד.

תשובה 14 מעמוד 23

```
10 CLS
20 FOR N=0 TO 15
30 COLOR, N : CLS
40 FOR T=1 TO 1000 : NEXT T
50 NEXT N
60 GO TO 20
```

תשובה 15 מעמוד 23

```
30 COLOR, N, 15-N : CLS
```

תשובה 16 מעמוד 24

יש לשנות:

```
40 SOUND 50+200*N, 10
```

תשובה 17 מעמוד 25

```
10 CLS
20 FOR S=0 TO 59
30 LOCATE 10, 20 : PRINT S
40 FOR T=1 TO 500 : NEXT T
50 NEXT S
```

תשובה 18 מעמוד 25

יש להוסיף:

```
60 GO TO 20
```

תשובה 19 מעמוד 26

יש לשנות:

```
40 FOR T=1 TO 785 : NEXT T
```

תשובה 20 מעמוד 26

יש להוסיף או לשנות:

```
15 FOR D=0 TO 59
30 LOCATE 10, 20 : PRINT D; ":"; S
55 NEXT D
```

תשובה 21 מעמוד 27

התשובה היא שלילית! אסור לכתוב בתוכנית לולאות מצטלבות (נסה!)

תשובה 22 מעמוד 29

```
5  CLS
10 FOR X=3 TO 12
20 LOCATE 2, X : PRINT "$"
30 NEXT X
```

__תשובה 23 מעמוד 29__

```
5  CLS
10 FOR Y=3 TO 7
20 LOCATE Y, 4 : PRINT "$"
30 NEXT Y
```

__תשובה 24 מעמוד 30__

```
10 CLS
20 FOR Y=4 TO 13
30 FOR X=4 TO 13
40 LOCATE Y, X : PRINT "$"
50 NEXT X
60 NEXT Y
```

__תשובה 25 מעמוד 31__

יש להוסיף או לשנות:

```
12 L=10
20 FOR Y=4 TO 3+L
30 FOR X=4 TO 3+L
```

__תשובה 26 מעמוד 32__

יש להוסיף או לשנות:

```
18 CLS
70 GO TO 12
```

__תשובה 27 מעמוד 33__

הקטע: "=4*7" הוא מחרוזת והקטע: 4*7 הוא ביטוי חשבוני.

__תשובה 28 מעמוד 33__

- המחרוזת "IBM" בנויה מ-3 סימנים.
- המחרוזת "*25/7Λ" בנויה מ-6 סימנים.

__תשובה 29 מעמוד 35__

השמות האלה אינם חוקיים עבור משתני-מחרוזת:

2A$, B3, C$$, C*$

__תשובה 30 מעמוד 39__

יש להוסיף או לשנות:

```
15  LOCATE 1,1
100 GO TO 15
```

__תשובה 31 מעמוד 65__

יש להשלים:

```
10 X=2
20 FOR N=1 TO 5
30 LOCATE 4, X : PRINT "$"
40 X=X+4
```

תשובה 32 מעמוד 41

```
1Ø FOR Y=2 TO 8 STEP 3
3Ø LOCATE Y, X : PRINT "$"
5Ø NEXT Y
```

תשובה 33 מעמוד 42

```
5  Y=5
1Ø CLS
2Ø FOR X=15 TO 5 STEP -1
3Ø LOCATE Y, X : PRINT "$"
4Ø Y=Y+1
5Ø NEXT X
```

תשובה 34 מעמוד 42

יש להוסיף:

```
25 COLOR X-5
```

תשובה 35 מעמוד 44

```
4Ø LOCATE Y+1, X-1 : PRINT "###"
```

תשובה 36 מעמוד 44

```
5Ø LOCATE Y+2, X-1 : PRINT "X □ X"
```

תשובה 37 מעמוד 44

יש להוסיף או לשנות:

```
7  LOCATE 1, 1
1Ø INPUT "Y="; Y
2Ø INPUT "X="; X
6Ø GO TO 7
```

תשובה 38 מעמוד 45

```
5  CLS
7  LOCATE 1, 1
1Ø FOR X=3 TO 18 STEP 5
2Ø Y=2
3Ø LOCATE Y, X : PRINT "O"
4Ø LOCATE Y+1, X-1 : PRINT "###"
5Ø LOCATE Y+2, X-1 : PRINT "X□X"
6Ø NEXT X
```

תשובה 39 מעמוד 45

יש להוסיף או לשנות:

```
2Ø FOR Y=2 TO 1Ø STEP 4
55 NEXT Y
```

תשובה 40 מעמוד 47

יש לשנות:

```
4Ø LOCATE Y+1, X-1 : PRINT CHR$(177);CHR$(177);
CHR(177)
```

תשובה 41 מעמוד 47

יש לשנות:

```
5Ø LOCATE Y+2, X-1:PRINT CHR$(221);"□";CHR$(222)
```

תשובה 42 מעמוד 47

```
10 CLS
20 INPUT "C="; C
30 LOCATE 10, 20 : PRINT CHR$ (C)
40 GO TO 20
```

תשובה 43 מעמוד 50

```
10 CLS
20 FOR X=2 TO 36
30 LOCATE 4, X : PRINT ">"
40 FOR T=1 TO 40 : NEXT T
50 LOCATE 4, X : PRINT "□ "
60 NEXT X
70 GO TO 20
```

תשובה 44 מעמוד 51

יש להוסיף או לשנות:

```
15 LOCATE 4, 1 : PRINT CHR$(219)
17 LOCATE 4, 37 : PRINT CHR$(219)
70 FOR X=36 TO 2 STEP -1
80 LOCATE 4, X : PRINT "<"
90 FOR T=1 TO 40 : NEXT T
100 LOCATE 4, X : PRINT " □ "
110 NEXT X
120 GO TO 20
```

תשובה 45 מעמוד 51

```
10 CLS
20 FOR X=20 TO 35 STEP 5
30 LOCATE 7, X : PRINT "*"
40 LOCATE 8, X : PRINT CHR$(219)
50 LOCATE 9, X : PRINT CHR$(219)
60 LOCATE 10, X-1 : PRINT CHR$(219); CHR$(219);
   CHR$(219)
70 NEXT X
80 FOR X=1 TO 39
90 LOCATE 7, X : PRINT ">"
100 FOR T=1 TO 40 : NEXT T
110 LOCATE 7, X : PRINT "□ "
120 NEXT X
```

תשובה 46 מעמוד 53

צלעות 1 ו-3 (שורות 20–50); צלעות 2 ו-4 (שורות 60–90)

```
10 CLS
20 FOR X=3 TO 10
30 LOCATE 3, X : PRINT "*"
40 LOCATE 7, X : PRINT "*"
50 NEXT X
60 FOR Y=4 TO 6
70 LOCATE Y, 3 : PRINT "*"
80 LOCATE Y, 10 : PRINT "*"
90 NEXT Y
```

תשובה 47 מעמוד 54

יש להוסיף או לשנות:

```
15 INPUT "L="; L
20 FOR X=3 TO 2+L
80 LOCATE Y, 2+L : PRINT "*"
```

תשובה 48 מעמוד 54

יש להוסיף או לשנות:

```
15 INPUT "NAME="; A$
17 LOCATE 5, 5 : PRINT A$
20 FOR X=3 TO 2+LEN (A$) + 4
80 LOCATE Y, 2+LEN (A$) + 4 : PRINT "*"
```

תשובה 49 מעמוד 54

יש להוסיף:

```
16 CLS
100 GO TO 100
```

תשובה 50 מעמוד 56

```
10 CLS
20 A=20 : B=20
30 FOR Y=1 TO 20
40 FOR X=A TO B
50 LOCATE Y, X:PRINT CHR$(219)
60 NEXT X
70 A=A-1 : B=B+1
80 NEXT Y
```

תשובה 51 מעמוד 56

יש להוסיף או לשנות:

```
5 INPUT A$
50 LOCATE Y, X : PRINT A$
100 GO TO 5
```

תשובה 52 מעמוד 56

יש לשנות:

```
70 B=B+1
```

תשובה 53 מעמוד 58

```
10 CLS
20 A=19 : B=21 : C=9
30 FOR X=A TO B
40 LOCATE C, X : PRINT CHR$(219)
50 NEXT X
```

תשובה 54 מעמוד 58

יש להוסיף או לשנות:

```
20 A=19 : B=21 : C=9 : D=11
60 FOR Y=C TO D
70 LOCATE Y, X : PRINT CHR$(219)
80 NEXT Y
```

תשובה 55 מעמוד 58

יש להוסיף:

```
90 FOR X=B TO A STEP -1
100 LOCATE D, X : PRINT CHR$(219)
110 NEXT X
```

תשובה 56 מעמוד 59

יש להוסיף:

```
25 FOR N=1 TO 4
120 FOR Y=D TO C STEP -1
130 LOCATE Y, A : PRINT CHR$(219)
140 NEXT Y
150 A=A-2 : B=B+2 : C=C-2 : D=D+2
160 NEXT N
```

משהו על היחידות הבאות

יחידה 3

ביחידת לימוד זאת הנקראת: "פיתוח משחקי מחשב טלויזיה", תעסוק בפיתוח תוכניות של משחקי מחשב-טלויזיה ותוך כדי כך תמשיך ותלמד מושגים נוספים בשפת ה-BASIC.

בין השאר תלמד על:

- הוראת התנאי: IF-THEN
- תהליכים אקראיים
- קליטת נתונים (READ-DATA)
- מערכים

בסיומה של יחידת לימוד זאת תוכל כבר לתכנת ברמה מתקדמת.

יחידה 4

יתכן שכבר ביחידה 1 התעורר אצלך הצורך לשמור תוכניות שכתבת על דיסקט.
לשם כך עליך להשתמש ב-DOS (קיצור ל-DISK OPERATING SYSTEM).
מהו DOS וכיצד תוכל לנצל אותו לצרכיך?

חוברת 4 של מחשבת הנקראת "PC-DOS" מוקדשת ללימוד רזי ה-DOS של ה-IBM-PC.

ושוב, כדרכה של מחשבת, לימוד ה-DOS יעשה בדרך פעלתנית - כפי שהדבר נעשה בשאר יחידות הלימוד של מחשבת.

לאחר שתסיים את יחידה 4 תכיר חלק מהפונקציות השונות של ה-DOS וכך תוכל לנצל ביתר יעילות את מגוון האפשרויות שמעמיד בפניך ה-IBM-PC.

יחידה 5

יחידה 5, הנקראת "גרפיקה עדינה", היא החוברת הראשונה בסידרה של 3 חוברות המוקדשות לפיתוח תוכניות בתחום הגרפיקה העדינה - HIGH RESOLUTION.
כדי ללמוד תחום מענין זה, המחשב שלך צריך להיות מצויד בכרטיס גרפי.
ביחידה 5 נגלה בפניך את עולמם של הפיקסלים (PIXELS) ובעזרתם נפתח תוכניות מגוונות, תוך כדי העשרת הידע התכנותי שלך. כמו כן נעסוק במעגלים, בצבעים ובעיפרון-וידיאו עדין.

יחידה 6

ביחידה 6, הנקראת "גרפיקה מתקדמת", נמשיך להעמיק את הידע שלך בכל הקשור לגרפיקה העדינה. ביחידה זאת נפתח תוכניות אשר ידרשו גם להקדיש מחשבה על מבנה התוכנית בעזרת תת-שגרות (SUBROUTINES).
כמו כן נעסוק בעיצוב אותיות-ענק ובעיצוב תמונות.

יחידה 7

יחידת הלימוד השביעית, הנקראת "גרפיקה מתמטית", מוקדשת לפיתוח תוכניות גרפיות מְרהיבות.
ביחידה זאת לא נוכל להתחמק משימוש בכמה מושגים מתחום הטריגונומטריה כמו הסינוס והקוסינוס. אל דאגה! אינך צריך להכיר מושגים אלה. כדרכנו, נלמד אותך צעד אחר צעד כל מושג מתמטי חדש שנעשה בו שימוש, ובעזרת מושגים אלו נקבל צורות מתוחכמות על המסך.

יחידה 8

יחידה 8, הנקראת "עיבוד נתונים", תצא לאור בינואר 1987.
חוברת זאת מוקדשת ללימוד השימוש בקבצים לצורך עיבוד נתונים.
את הידע שתרכוש תנצל על מנת לבנות מדריך טלפון ממוחשב, שמירת
תמונות בעזרת קבצים ופיתוח ספרייה ממוחשבת.

לוגו עברית ★★★ LOGO ★★★ לוגו עברית

בינואר 1987 תצא לאור ערכת לוגו של מחשבת עבור IBM-PC
ותואמיו בגירסה עברית.
ערכת הלוגו של מחשבת, הנקראת "עולמו של הצב", מכילה תוכנה
מיוחדת אשר מאפשרת לכתוב תוכניות בשפת התכנות לוגו (LOGO)
בשפה העברית, דבר אשר יאפשר ללמוד את יסודות שפת ה-LOGO מבלי
שהשפה האנגלית תהווה מחסום.
הערכה מכילה ספר אשר בעזרתו תלמד להניע את הצב המפורסם של
הלוגו על גבי המסך, ובעזרתו תשרטט שרטוטים מדהימים ביופיים
על המסך.
כדרכה של מחשבת תעשה זאת בדרך פעלתנית, תוך כדי ביצוע משימות
מגוונות, ותוך כדי כך תלמד את עקרונות התכנות בשפת הלוגו.

ציורי מחשבת

במרץ 1987 תצא לאור ערכת **"ציורי מחשבת"** עבור IBM-PC ותואמיו:

"ציורי מחשבת" תאפשר לכל אחד לצייר על פני כל המסך ציורים מאד מתוחכמים.

ציורי מחשבת מכילה:

1. דיסקט המכיל מחולל גרפיקה המשוחח בשפה העברית, אשר מאפשר בין השאר ובאופן ידידותי מאוד:

- לצייר על פני כל המסך.
- ליצור דמויות כמו אנשים, חיות, מטוסים ועוד, אותם ניתן להגדיל ולהציג בכל מקום על פני המסך.
- לשמור את הציורים והדמויות על דיסקט.

2. חוברת לימוד אשר בעזרתה:

- תלמד לנצל את כל האפשרויות שמחולל הגרפיקה מעמיד לרשותך.
- תלמד את יסודות הפרספקטיבה בציור.
- תלמד טכניקות מיוחדות, אשר בעזרתן תוכל להפוך ציור רגיל לציור ממוחשב.

לאחר שתלמד לנצל את כל הכלים שהמחולל מעמיד לרשותך תוכל לצייר ציורים מאוד מתוחכמים כיד הדמיון הטובה עליך.
דע לך, כי הציורים שתיצור ישמשו לעוד מטרה: הם ישמשו כרקע בסרטי האנימציה שתיצור עם מחולל האנימציה של מחשבת... (על כך בעמוד הבא).
הנה דוגמא לאחד הציורים שנעשו בעזרת המחולל:



סרטי מחשבת

באפריל 1987 תצא לאור ערכת **"סרטי מחשבת"** עבור IBM-PC ותואמיו.

"סרטי מחשבת" תאפשר לכל אחד ליצור סרטי אנימציה (הנפשה) כמו אלו המוצגים בטלויזיה ובמשחקי המחשב המתוחכמים. לא נדרש שום רקע בתכנות או ידע קודם בעקרונות סרטי האנימציה.

"סרטי מחשבת" מכילה:

1. דיסקט המכיל מחולל ואנימציה המשוחח בשפה העברית, אשר מאפשר, בין השאר, ובאופן ידידותי מאוד:

- ליצור דמויות כמו אנשים, חיות, מטוסים ועוד ועוד.
- להניע את הדמויות על המסך באופן כזה שיתקבל סרט אנימציה.
- לשמור את כל הציורים והסרטים על הדיסקט, וכך תוכל להציג את הסרט שתיצור בכל עת שתרצה.
- לשלב מנגינות בסרט.

2. חוברת לימוד אשר בעזרתה:

- תלמד לנצל את כל האפשרויות שמחולל הגרפיקה והאנימציה מעמיד לרשותך.
- תלמד את עקרונות סרטי האנימציה כך, שתוכל לביים וליצור בעצמך סרטי אנימציה.

כדרכה של מחשבת תעשה זאת בדרך פעלתנית תוך כדי ביצוע משימות מגוונות וכך, אנו מקווים, פיתוח סרטי אנימציה יהפוך לחויה יצירתית ממדרגה ראשונה.

שים לב!

תוכל לשלב בסרט ציורי רקע שתיצור בעזרת התוכנה של "ציורי מחשבת" (ראה עמוד קודם). וכך לדוגמא תוכל לגרום למטוס לחלוף בשמים, כאשר ברקע מרחפים עננים.